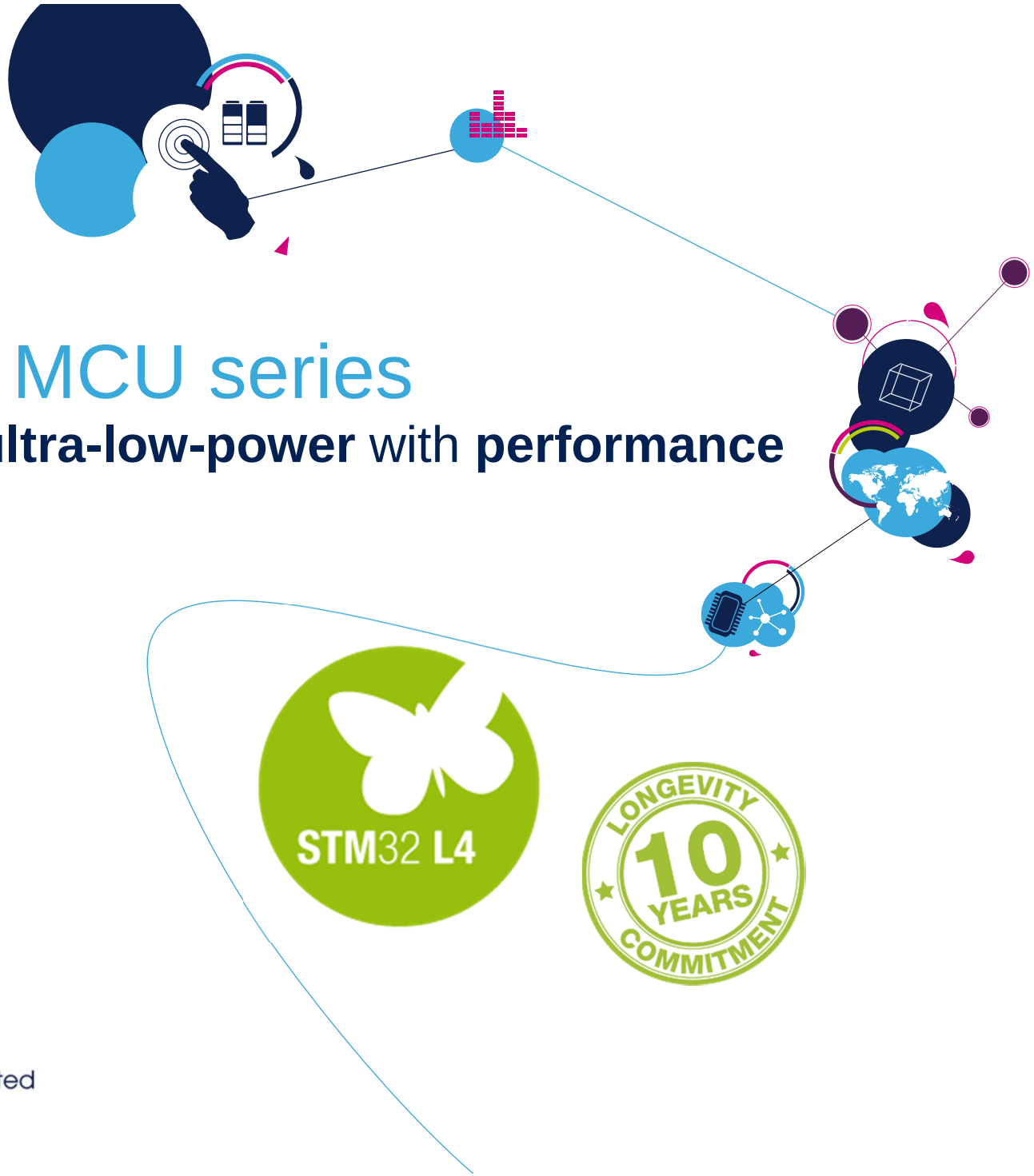
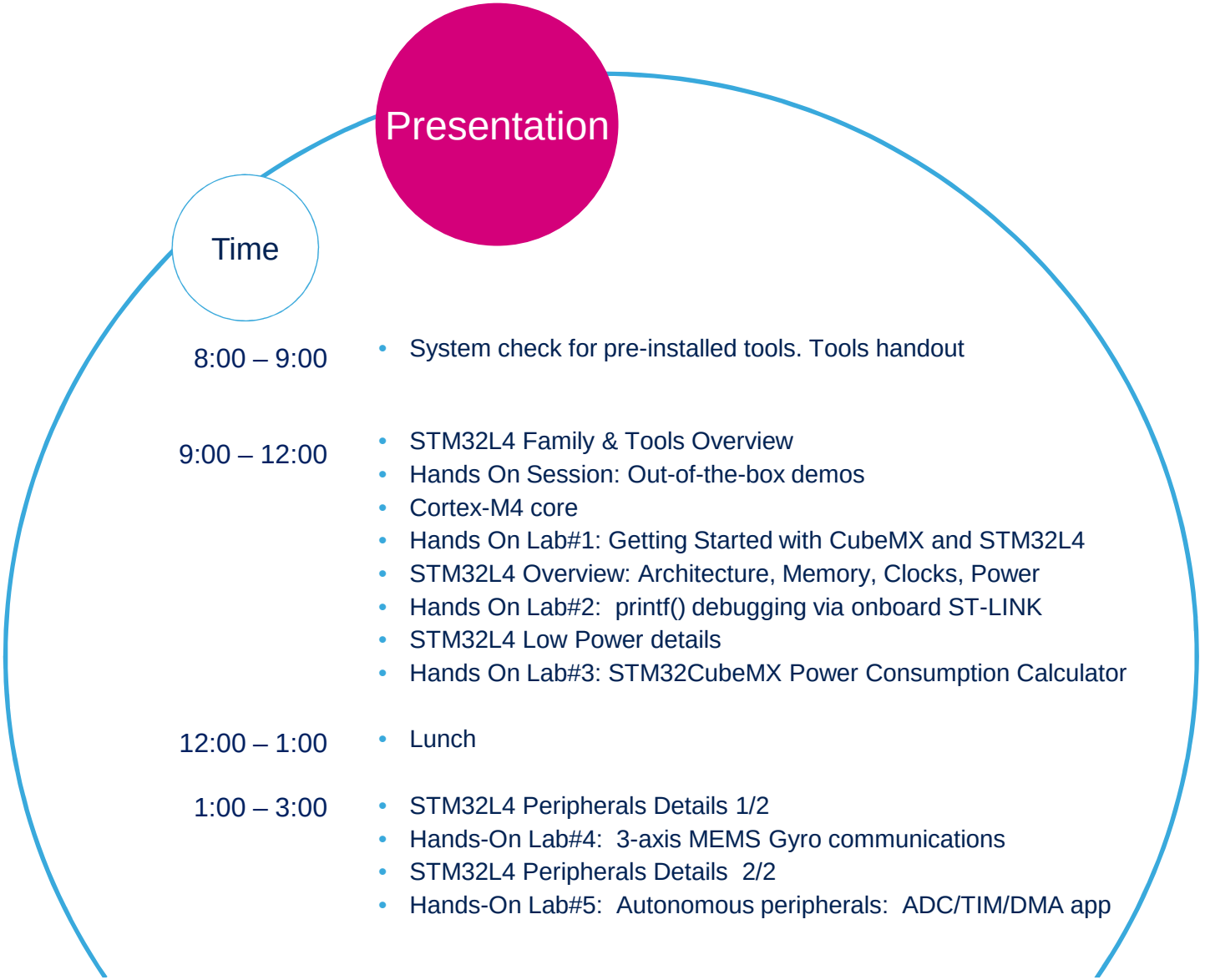


STM32L4 MCU series

Excellence in **ultra-low-power** with **performance**

life.augmented

Agenda

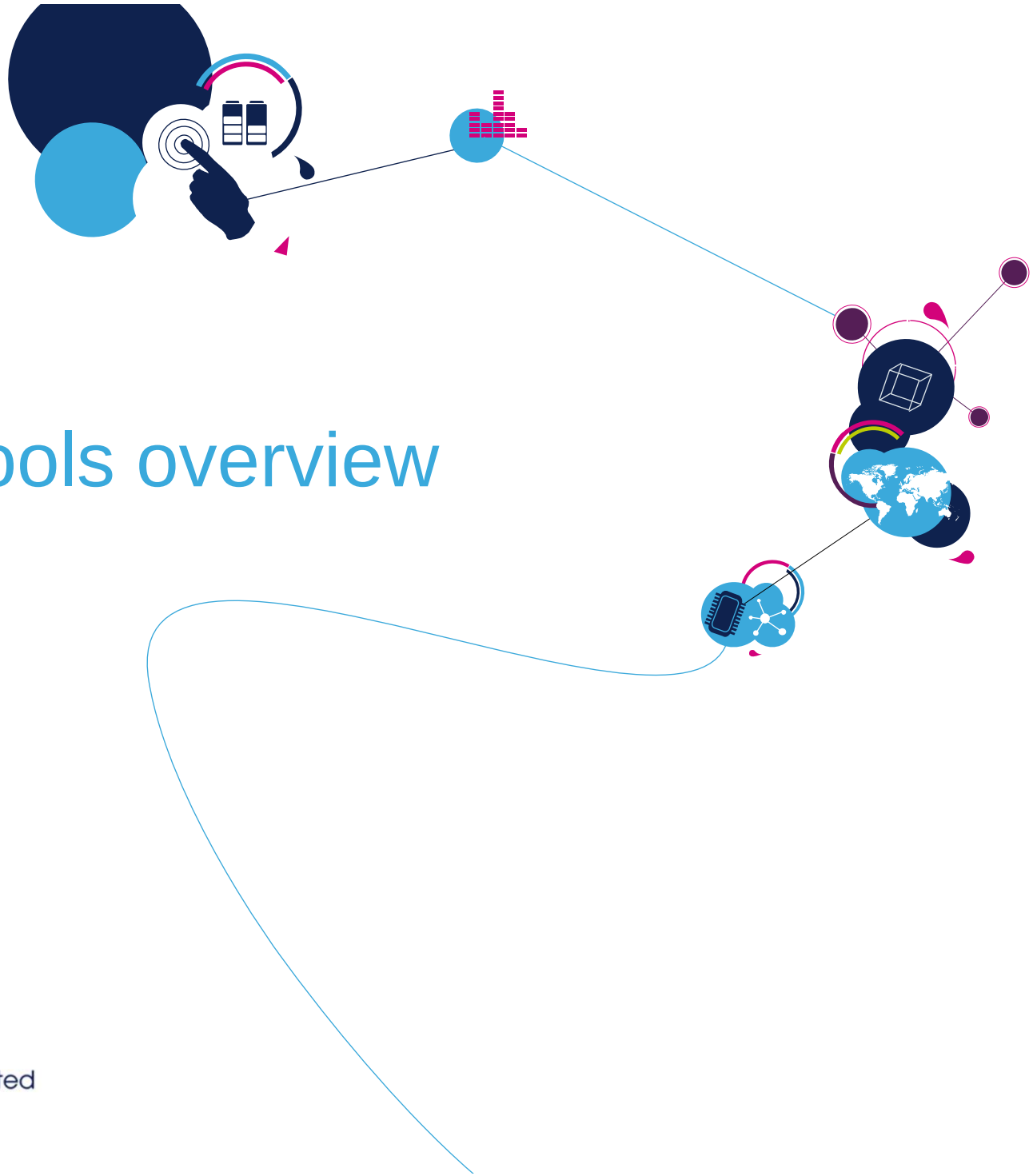

Presentation
Time
8:00 – 9:00
System check for pre-installed tools. Tools handout
9:00 – 12:00
STM32L4 Family & Tools Overview
Hands On Session: Out-of-the-box demos
Cortex-M4 core
Hands On Lab#1: Getting Started with CubeMX and STM32L4
STM32L4 Overview: Architecture, Memory, Clocks, Power
Hands On Lab#2: printf() debugging via onboard ST-LINK
STM32L4 Low Power details
Hands On Lab#3: STM32CubeMX Power Consumption Calculator
12:00 – 1:00
Lunch
1:00 – 3:00
STM32L4 Peripherals Details 1/2
Hands-On Lab#4: 3-axis MEMS Gyro communications
STM32L4 Peripherals Details 2/2
Hands-On Lab#5: Autonomous peripherals: ADC/TIM/DMA app

life.augmented

Systems Check

- Everyone should have:
  - A Laptop running Windows OS (XP, Vista, Win7, Win8 or Win10)
  - USB Cable
  - USB Flash Drive
  - STM32L476G-DISCO Discovery kit

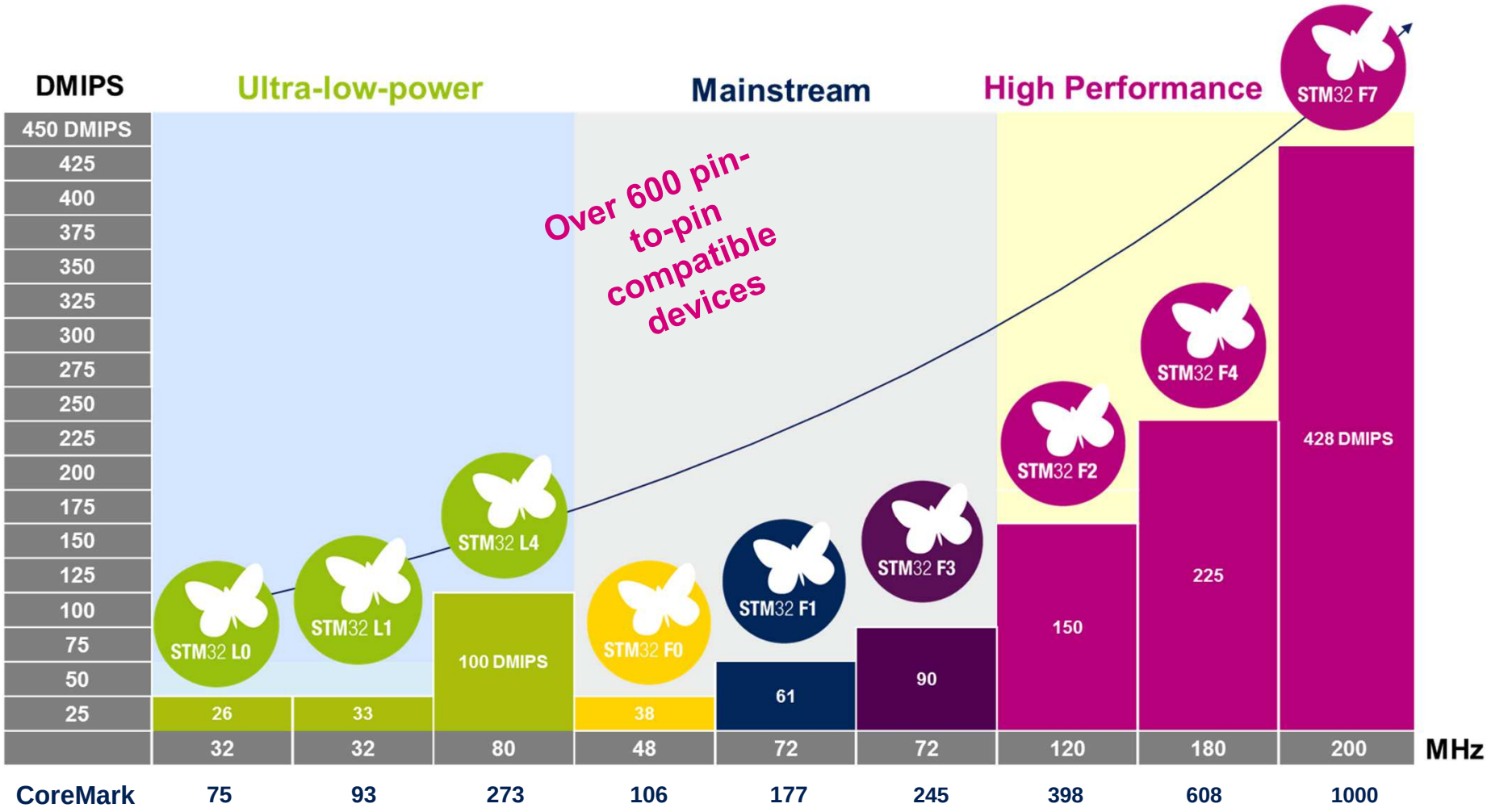
- The following software tools installed:
  - STM32CubeMX, v4.10.0 + CubeL4 HAL Library, v1.0.0
  - ST-LINK Utility, v3.7.0
  - IAR Embedded Workbench for ARM, v7.40.5.9739
  - Teraterm (or equiv) terminal emulator



STM32L4
Family & Tools overview

life.augmented

Broadest 32-bit MCU product portfolio



life.augmented

Broad Range of Development Tools

STM32 Nucleo	Discovery kits	Evaluation boards	3rd parties
Flexible prototyping	Key feature prototyping	Full feature evaluation	From full evaluation to open hardware

www.st.com/stm32nucleo

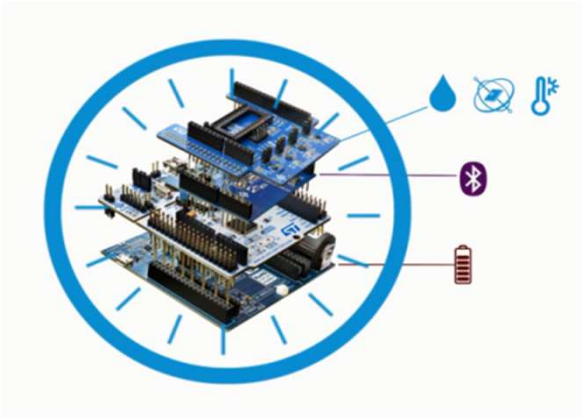
www.st.com/stm32discovery

STM32 Nucleo expansion boards

Specialized functionality add-on

Connectivity, Sensors…

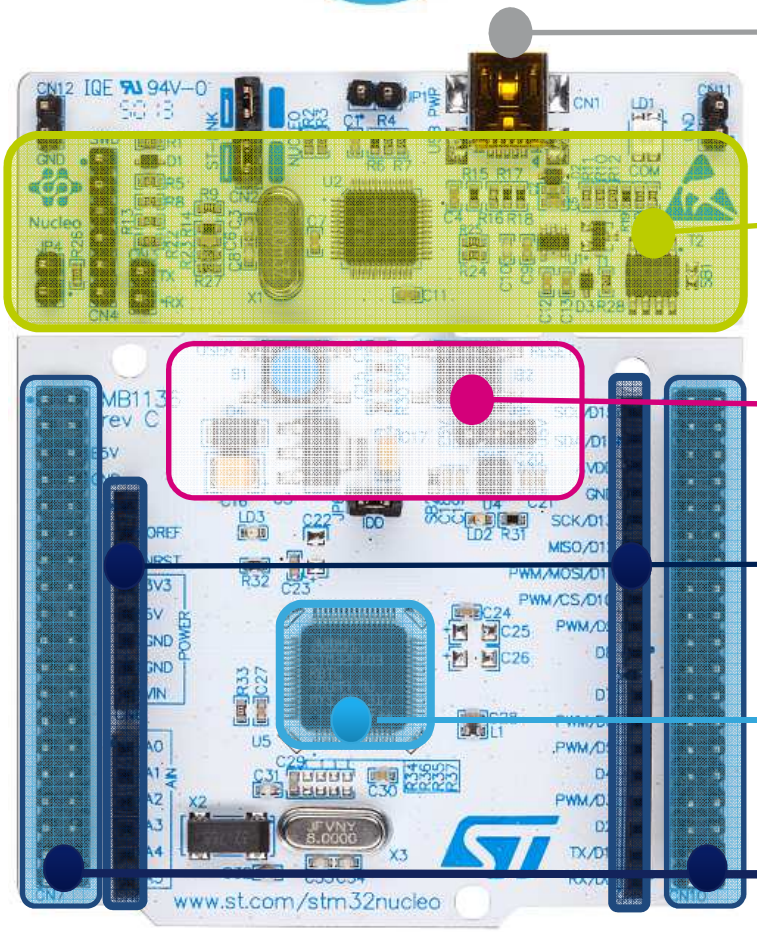
www.st.com/x-nucleo



life.augmented

NEW

STM32 Nucleo features



Flexible board power supply
Through USB or external source

Integrated ST-Link/V2.1
Mass storage device flash programming
Virtual COM port for communications

2 push buttons, 2 color LEDs

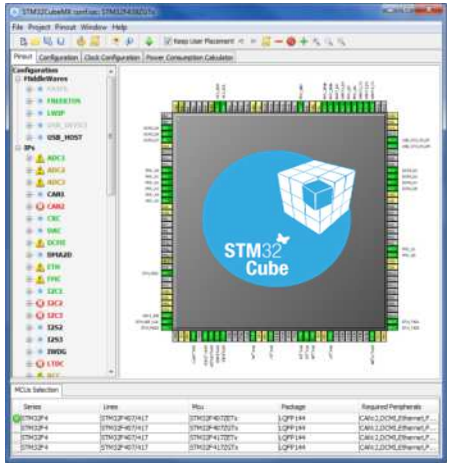
Arduino™ extension connectors
Easy access for add-ons

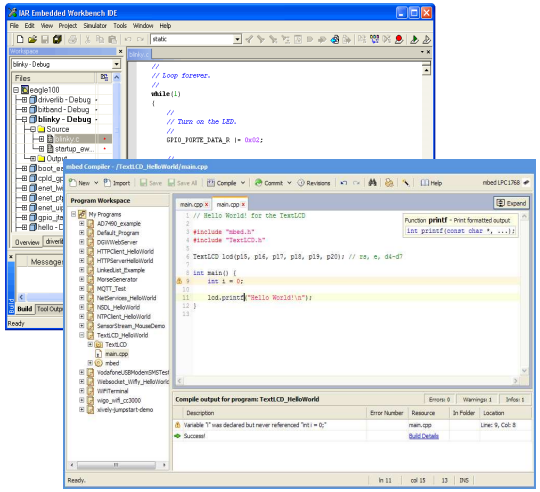
One STM32 MCU flavor with 64 pins

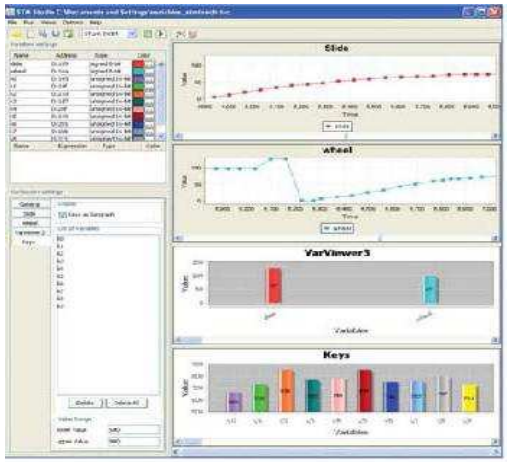
Morpho extension headers
direct access to all MCU I/Os

life.augmented

Comprehensive choice of IDEs







STM32CubeMX — Generate Code

Partners IDEs — Compile & Debug

STMStudio — Monitor

IAR SYSTEMS · atollic · SYSTEM · EMPROG · TASKING · KEILABS · SEGGER Embedded Studio

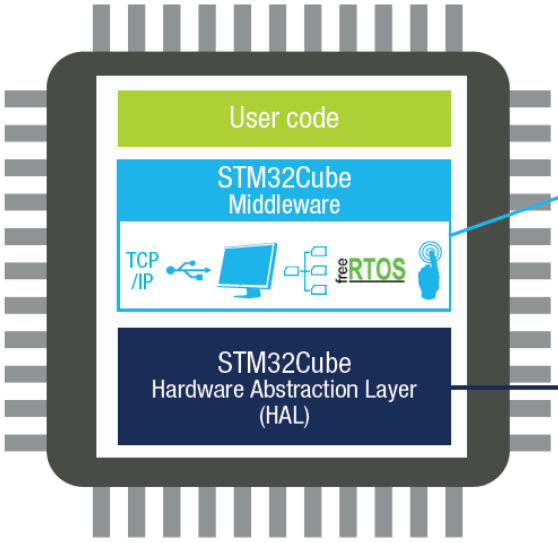
ARM KEIL Microcontroller Tools · CooCox · mbed · ac6

Free IDE

life.augmented

STM32Cube
Supporting all STM32 MCUs

- Generate your configuration code with the STM32Cube and you can focus on your added-value software !
  - 4 configuration wizards: pinout, clock, peripherals & middleware, power consumption
  - Portable Hardware Abstraction layer, from series to others
  - Middleware with RTOS, USB, TCP/IP, File System, Graphics , Touch sensing…



- Open-source TCP/IP stack (lwIP)
- USB Host and Device library from ST
- STemWin graphical stack library from ST and SEGGER
- Open-source FAT file system (FatFs)
- Open-source real-time OS (FreeRTOS)
- Dozens of examples

- Abstraction of STM32 MCU through portable APIs
- High coverage for most STM32 peripherals
- Production-ready using CodeSonar® static analysis tool
- Hundreds of examples
- Open-source BSD license

life.augmented

www.st.com/stm32cube

Comprehensive choice of STM32 free IDEs

System Workbench	CoIDE	MDK-ARM	
Microsoft	Microsoft	Microsoft	Microsoft
COMING			
COMING			
Free licenses for all STM32 microcontrollers		**Free licenses for STM32F0 & STM32L0**	**Free access to all STM32Nucleo users**

life.augmented

New **ST MCU Finder** Application

ANDROID APP ON Google play

Available on the App Store

Windows Store

- Quickly find the right ST MCU
- Easy access to technical materials
- Latest news from ST MCU world

www.st.com/stmcufinder

life.augmented

Birth of the STM32 L4

High-performance
ARM Cortex-M4 FPU, DSP

Advanced analog,
New digital peripheral set

Ultra-low-power

- STM32L4 is a perfect fit in terms of ultra-low power, performances, memory size, and peripherals at a cost effective price.
- Convergence between High performance and Ultra-low power series

life.augmented

Key messages of STM32 L4 series

1. **ULP leader and performance booster** ST has built a new architecture delivering best-in-class, ultra-low-power (ULP) figures thanks to its high flexibility. In addition the performance of the STM32L4 far exceeds the competition in the ultra-low-power world. It delivers 100 DMIPS based on its ARM Cortex-M4 core with FPU and ST ART Accelerator™ at 80 MHz.

2. **Innovation** Covering a large range of applications the STM32L4 features many architectural innovations and new smart embedded peripherals.

3. **Integration and safety** 1 MB of Flash and 128 KB of SRAM with safety and security features, smart and numerous peripherals, advanced and low power analog circuits in packages as small as 3.8 x 4.4 mm.

4. **Great Investment** This new STM32 member benefits from the pin-to-pin compatibility of the STM32 family and the STM32 Ecosystem.

life.augmented

Ultra-Low-Power and Flexibility

FlexPowerControl

STM32L4 is based on a new platform optimized to reduce power consumption and increase flexibility

Down to 30 nA for I/O wake-up
with additional Shutdown mode

RTC available for all power modes
(from Active down to V_{BAT})

External level shifter no longer needed
Separate V_{DD} supplies
(down to 1.08 V)

4 nA V_{BAT} mode with charging capability
Automatic switch to maintain power for RTC and backup registers

Down to 360 nA keeping 32 Kbytes of SRAM active
in Standby mode

Wake up MCU with any peripheral
(Communication I/Fs, analog circuits, timers…)

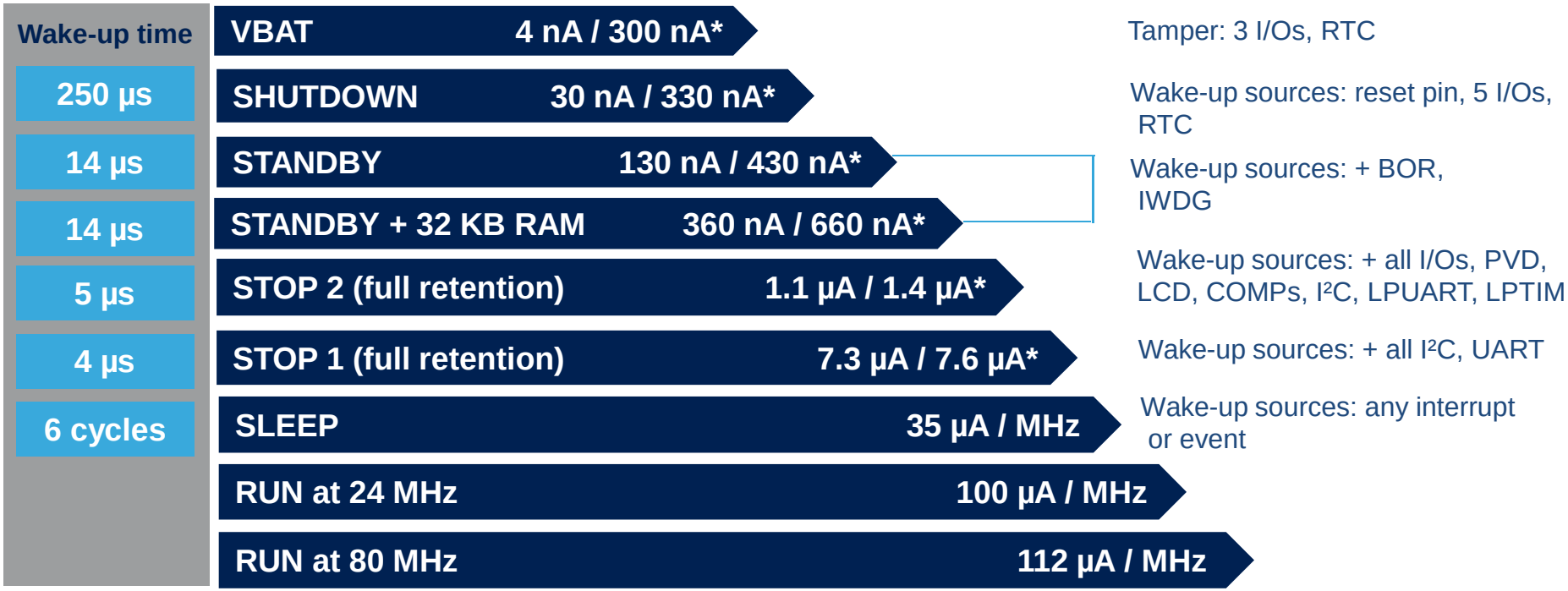
USB capable with 32 kHz crystal oscillator
(Dedicated crystal oscillator is no longer needed for USB function)

I/O level kept in low power modes
Optimization of system consumption

Internal oscillator from 100 kHz to 48 MHz
(+/-0.25% int. clock accuracy over voltage/temperature with LSE)

Ultra-low-power modes

Best power consumption numbers with full flexibility



Note : * without RTC / with RTC

STM32L4

This ULPBench® winner takes off like a rocket

Dhrystone MIPS 100

ULPBENCH™ An EEMBC Benchmark 153

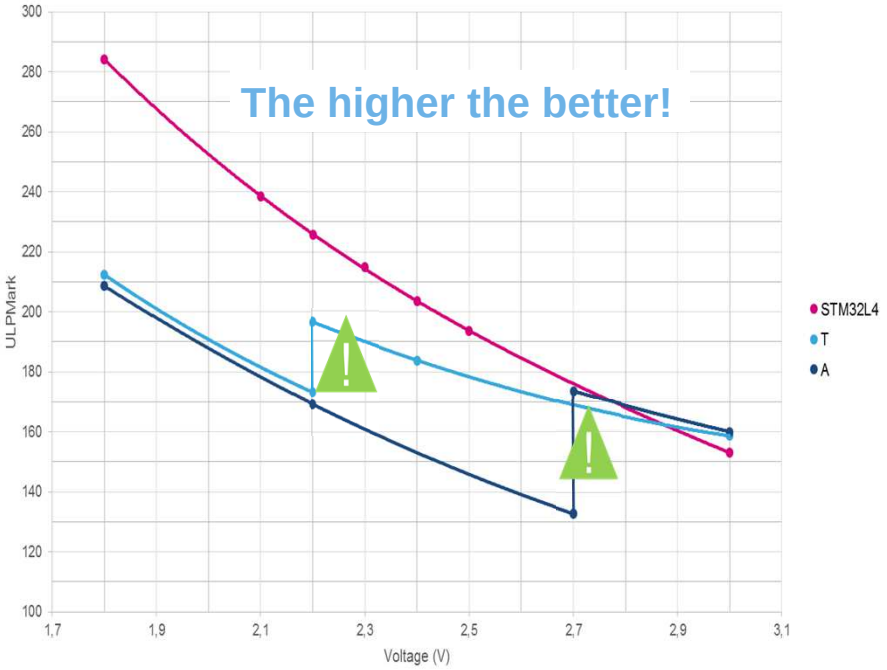
COREMARK® An EEMBC Benchmark 273

www.st.com/stm32l4

From 0 to 48 MHz in less than 5 µs
From 0 to 80 MHz in less than 20 µs

life.augmented

And shatters performance limits in ULP world

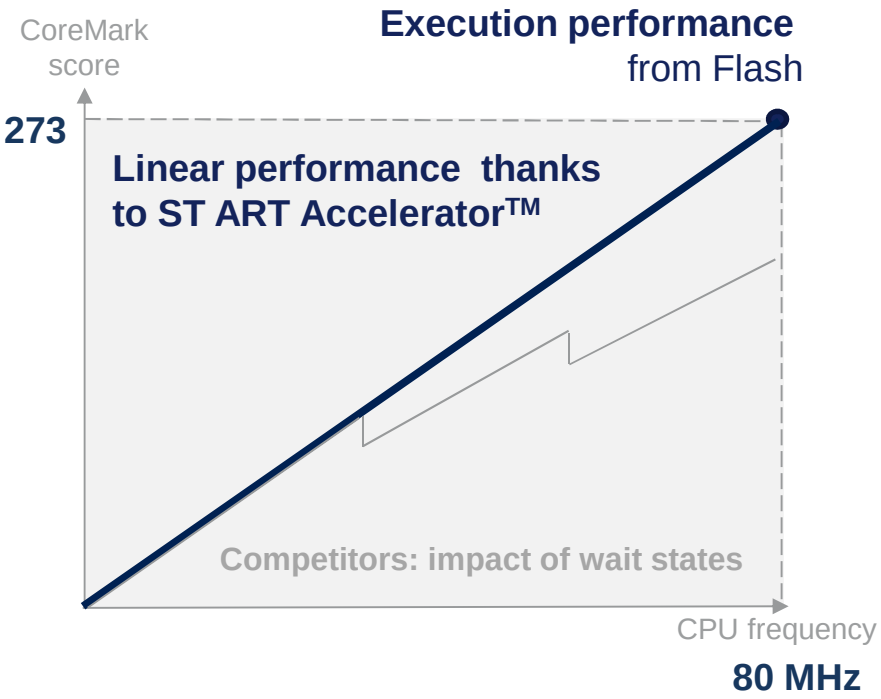


On competition devices: discontinuity due to lack of DC/DC functionality when voltage decreases

No external coil and capacitor required for STM32L4

Providing more performance

Do not compromise on performance with STM32L4



- **Up to 80** MHz/ **100 DMIPS** with ART Accelerator™
- **Up to 273** CoreMark Result
- ARM Cortex-M4 with DSP instructions and floating-point unit (FPU)
- Optimized DMA (14 channels)
- SPI up to 40 Mbit/s, USART 10 Mbit/s

High integration

High integration with high memory size in small packages

Parallel Interface
FSMC 8-/16-bit (TFT-LCD, SRAM, NOR, NAND)

Display
LCD driver 8 x 40

Timers
17 timers including:
2 x 16-bit advanced motor control timers
2 x ULP timers
7 x 16-bit-timers
2 x 32-bit timers

I/Os
Up to 114 I/Os
Touch-sensing controller

Cortex-M4
80 MHz
FPU
MPU
ETM

DMA

ART Accelerator™

Up to 1-Mbyte Flash with ECC Dual Bank

128-Kbyte RAM

Connectivity
USB OTG,
1x SD/SDIO/MMC, 3 x SPI,
3 x I²C, 1x CAN,
1 x Quad SPI,
5 x USART + 1 x ULP

Digital
AES (256-bit), TRNG, 2 x SAI, DFSDM (8 channels)

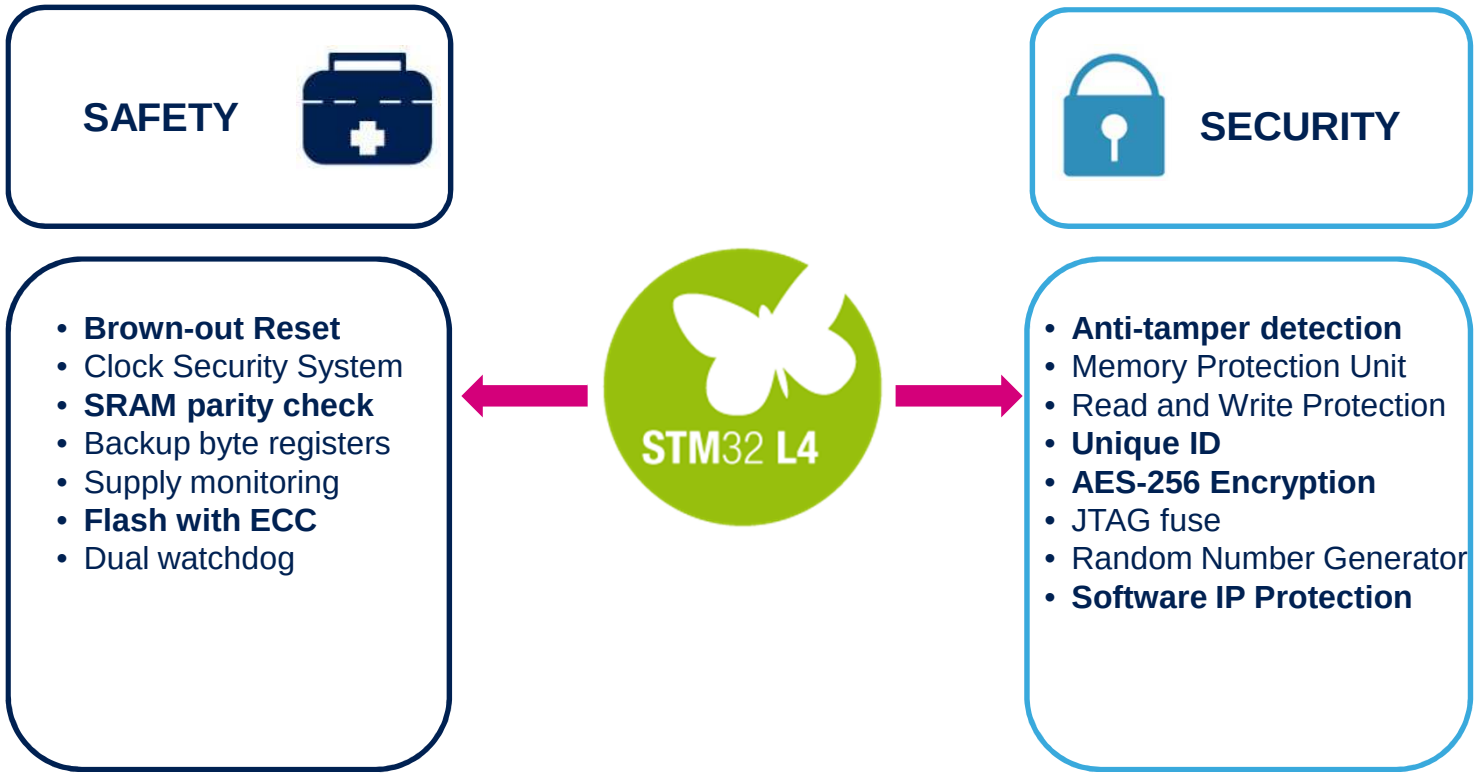
Analog
3 x 16-bit ADC, 2 x DAC,
2 x comparators,
2 x op amps
1 x temperature sensor

Package size down to 4.4 x 3.8 mm

life.augmented

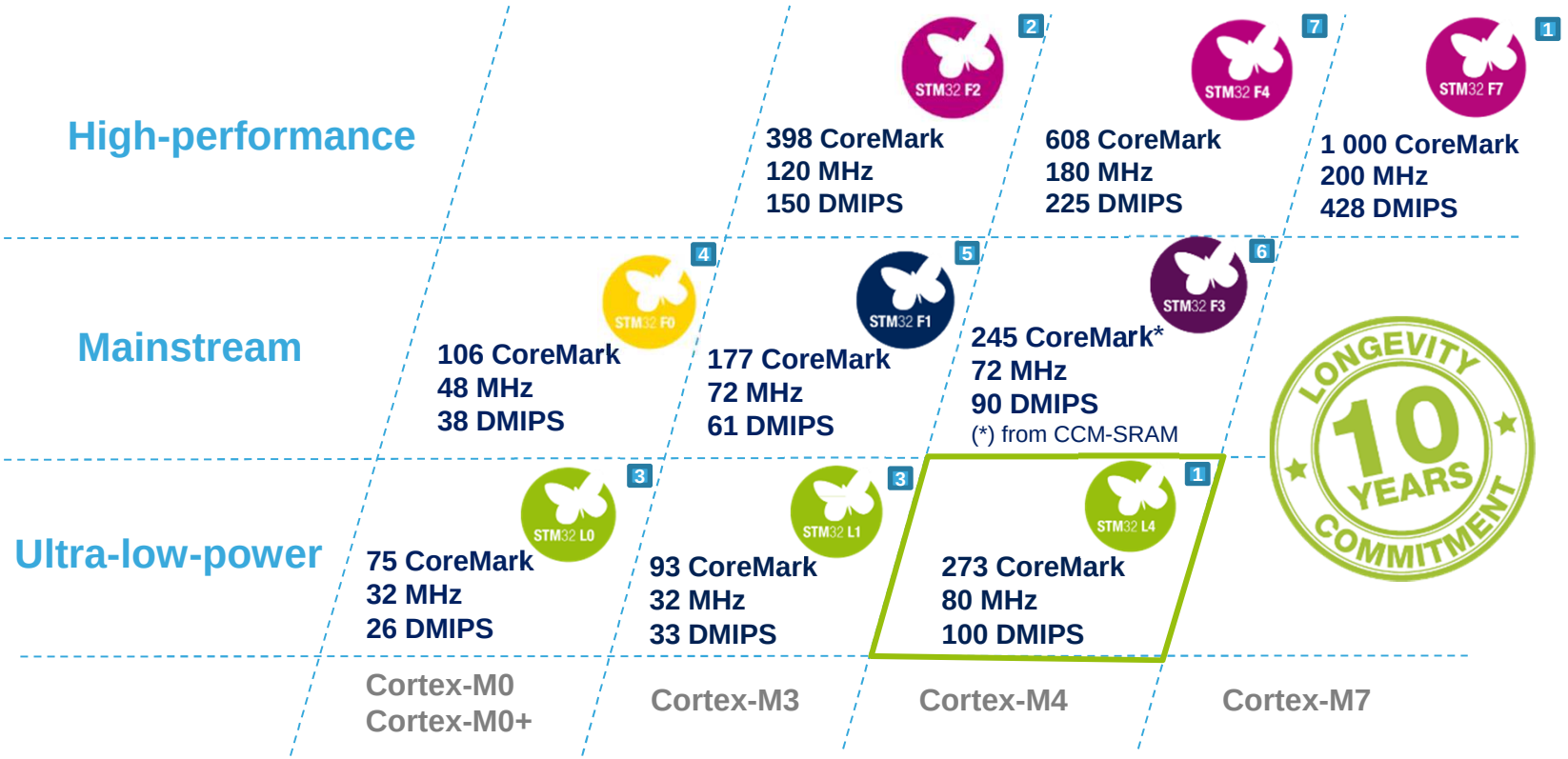
Safety and security

Integrated safety and security features



STM32L4: continuity in STM32 portfolio

9 product series / 32 product lines
STM32L4 benefits from pin-to-pin compatibility across the family



life.augmented

number of lines

4 Great investment

STM32L Ultra-Low-Power offering

STM32L4 completes the ultra-low-power family

Cost-smart ULP Champion

Cortex-M0+ @ 32 MHz
Operating range:
1.65 to 3.6V
8/16-bit applications
Numerous pin counts

3 product lines,
Cost-effective,
Smaller packages
USB, LCD, Analog
16 to 192 Kbytes of Flash
Up to 20 Kbytes of SRAM

Broad Range Foundation

Cortex-M3 @ 32 MHz
Operating range:
1.65 to 3.6V
Wide choice of memory sizes

3 product lines,
USB, LCD, AES,
Rich Analog
True EEPROM,
Dual bank Flash (RWW)
32 to 512 Kbytes of Flash
Up to 80 Kbytes of SRAM

ULP with performance

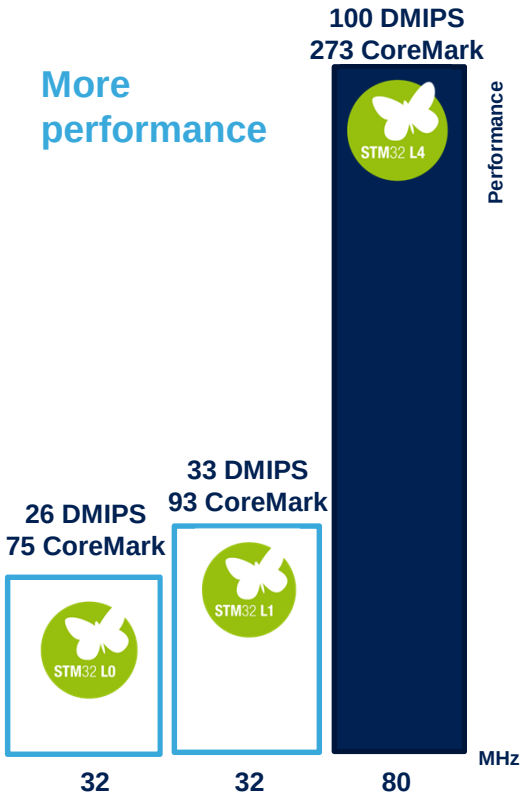
Cortex-M4 w/ FPU @ 80 MHz
Operating range:
1.71 to 3.6V
Advanced Peripheral Performance

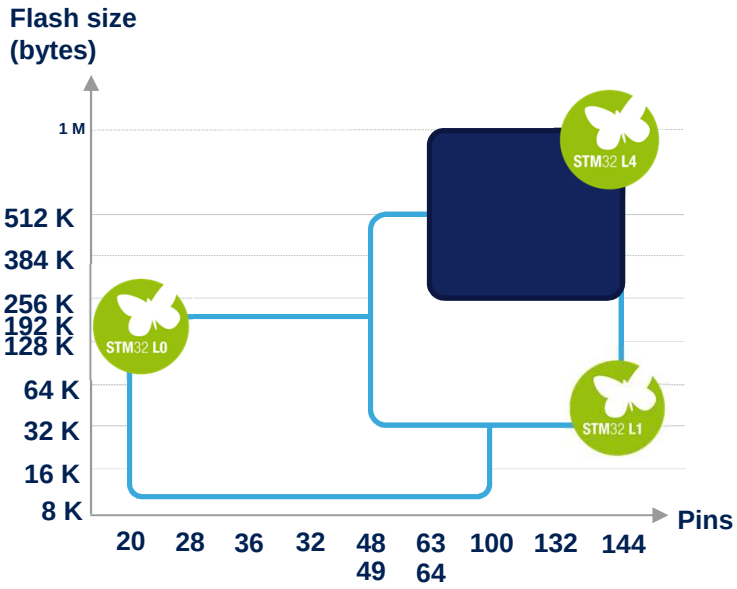
3 product lines,
ADC 5 Msps, PGA, Compar.,
DAC, op amp, USB OTG,
LCD, AES
256 Kbytes to 1 Mbyte of Flash
Up to 128 Kbytes of SRAM

life.augmented

STM32L, a complete offering

STM32L4 completes the ultra-low-power family





STM32L4 series

Cortex®-M4 (DSP + FPU) – 80 MHz

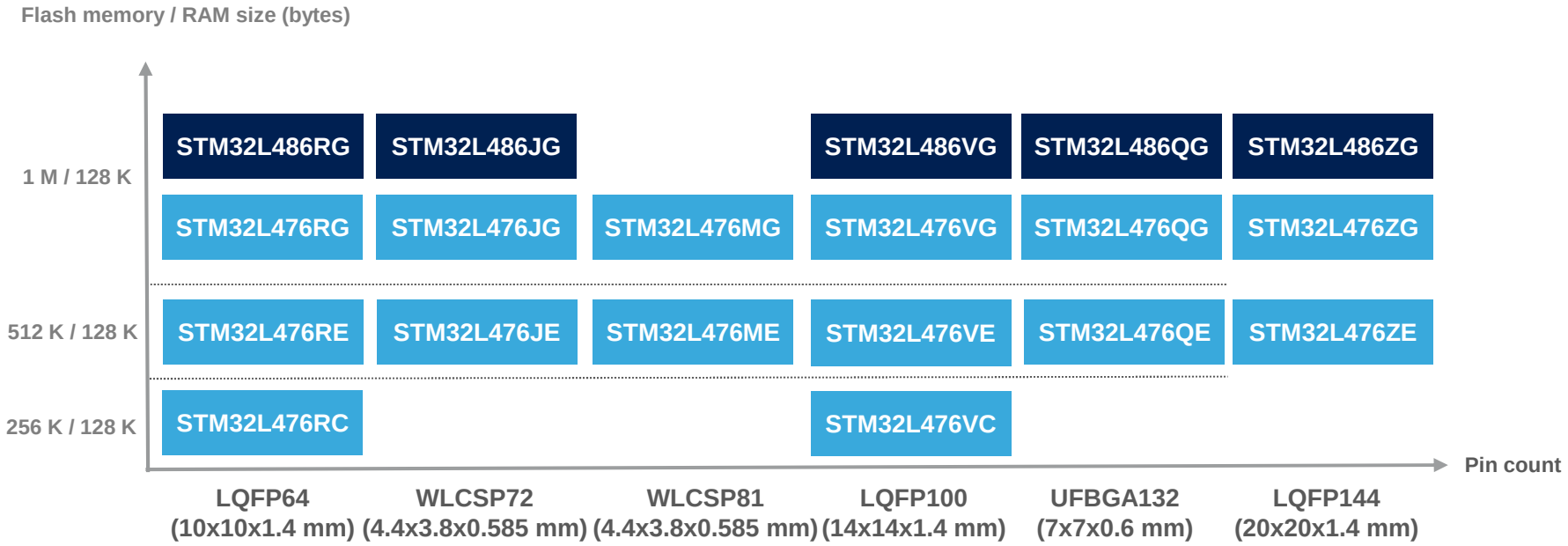
- ART Accelerator™
- USART, SPI, I²C
- Quad SPI
- 16 and 32-bit timers
- SAI + audio PLL
- SWP
- 1x CAN
- 2x 12-bit DAC
- Temperature sensor
- Low voltage 1.71 to 3.6 V
- V_{BAT} Mode
- Unique ID
- Capacitive Touch sensing

Product line	Flash (KB)	RAM (KB)	Memory I/F	2 x Op amps	2 x Comp	8-ch / 4x Sigma Delta Interface	16- bit ADC (5 Msps)	USB2.0 OTG FS	Segment LCD Driver	AES 128/256-bit
STM32L471 Access	512 to 1024	128	SDIO FSMC	●	●	●	3			
STM32L475 USB OTG	256 to 1024	128	SDIO FSMC	●	●	●	3	●		
STM32L476 USB OTG & LCD	256 to 1024	128	SDIO FSMC	●	●	●	3	●	Up to 8x40	
STM32L486 USB OTG & LCD & AES	1024	128	SDIO FSMC	●	●	●	3	●	Up to 8x40	●

Legend: Available in Q4/2015

life.augmented

STM32L4 portfolio



Legend:

With 128/256-bit AES hardware encryption

Without encryption

life.augmented

STM32L4 ecosystem

HARDWARE TOOLS

STM32 Nucleo

Flexible prototyping

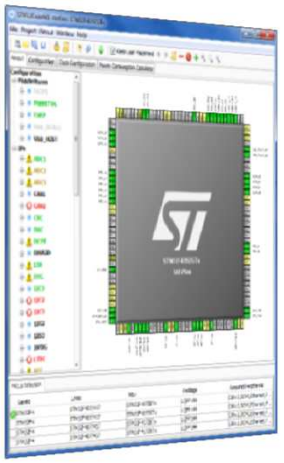
Discovery kit

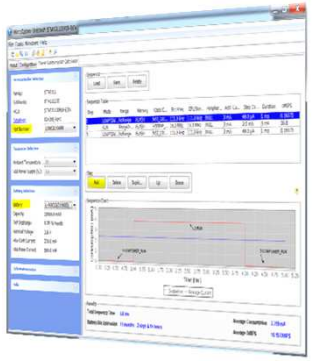
Key feature prototyping

Evaluation board

Full feature evaluation

SOFTWARE TOOLS





STM32CubeMX featuring code generation and power consumption calculation

STM32 ODE
Nucleo and X-Nucleo

The building blocks	Your need	Our answer

Accelerometer, gyroscope
Inertial modules, magnetometer
Pressure, temperature, humidity, UV
Proximity, microphone

Sense — DATA COLLECT

Bluetooth LE, Sub-GHz radio
NFC, Wi-Fi, GNSS

Connect — DATA TRANSMIT

Audio amplifier
Touch controller
Operation Amplifier

Translate — DATA ACCESS

Stepper motor driver
DC & BLDC motor driver

Move / actuate — DATA CREATE

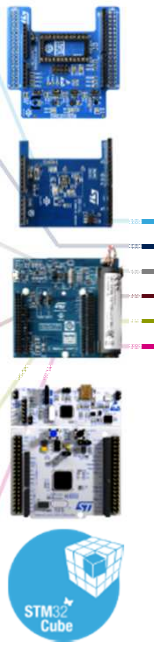
Energy management & battery

Power — DATA POWER

General purpose microcontrollers
Secure microcontrollers

Process — DATA PROCESS

Software



STM32 Nucleo
Nucleo Expansion Boards

life.augmented

Hands-On Session:
Out-of-the-box demos

life.augmented

STM32L476 Discovery – HMI

Integrated ST-Link/V2-1 (for programming and debugging)

LCD 96 segments

Motion Mems (9-axis)

push buttons and joystick,
2 color LEDs

Quad SPI NOR Flash
16 MB

USB OTG connector

STM32L476 Discovery - Audio and connector

NFC / ACP connector

MFX to auto-measure power consumption

Direct access to all MCU I/Os

Audio Codec and 3.5 mm connector

Microphone Mems

life.augmented

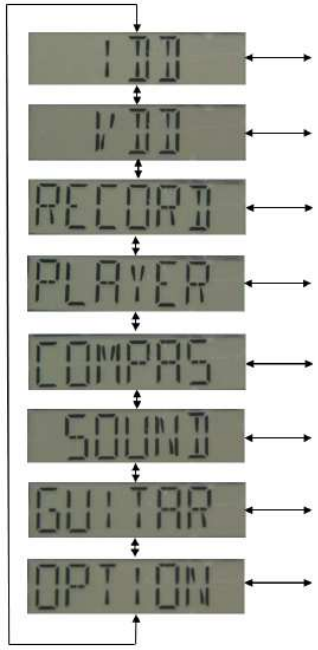
STM32L476 Discovery - back side

Flexible board power supply
CR2032 battery or USB

Out-of-the-Box Demos

- Plug the USB cable into the top Mini-B connector of the Discovery Board

- Use the blue joystick to scroll through the application menu:
  - IDD Measurement app
  - VDD Measurement app
  - Record / Playback app
  - Compass app
  - Sound Meter app
  - Guitar Tuner app
  - Options

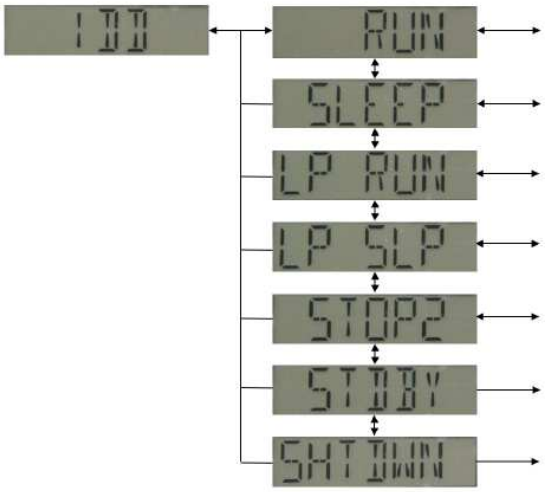


- The project is in the CubeL4 library, Discovery Demonstrations folder:
  - C:\Users\xxxx\STM32Cube\Repository\STM32Cube_FW_L4_V1.0.0\Projects\STM32L476G-Discovery\Demonstrations\EWARM

life.augmented

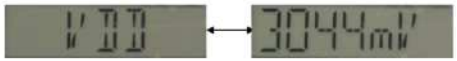
IDD / VDD Measurement apps

Mode	Description
Run	Run 24Mhz, voltage range 2, PLL off, RTC/LSE off, Flash ART on
Sleep	Sleep 24Mhz, voltage range 2, PLL off, RTC/LSE off, Flash ART on
Low-power run	Low Power Run 2Mhz, PLL off, RTC/LSE off, Flash ART on
Low-power sleep	Low Power Sleep 2Mhz, PLL off, RTC/LSE off, Flash ART on
Stop 2	RTC/LSE off, Flash ART off
Standby	RTC/LSE off, Flash ART off, RAM retention off
Shutdown	RTC/LSE off, Flash ART off



The “IDD Measure” app uses the “MFX” onboard STM32L151 as a non-intrusive auto-ranging current measurement probe.

The target L4 will enter a low power state, the MFX will measure the current, wake the L4 up and report the measurement back over I2C.



life.augmented

Audio demonstrations

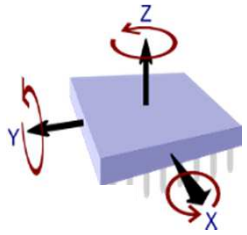
- RECORD application
  - Uses MP34DT01 MEMS microphone (LED5 toggling during record)
  - 16-bit audio samples @ 48 kHz stored in N25Q128A13 QuadSPI Flash
  - LEFT key to exit

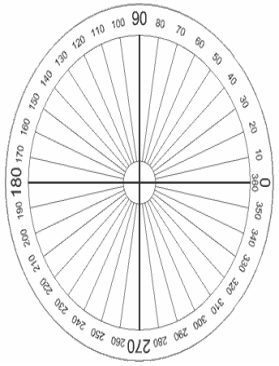
- PLAYER application
  - Uses CS43L22 audio DAC and 3.5mm jack output
  - Audio playback either from internal or QuadSPI Flash after a RECORD. Sub-menus :
    - “FLASH” : Audio playback of any WAV binary file loaded @ 0x08020000
    - “QSPI” : Audio playback from QuadSPI Flash
  - Options :
    - SEL key to pause/resume playback
    - UP/DOWN keys to control volume
    - Audio is played back in loops until LEFT key is pressed

life.augmented

Compass and sound meter demonstrations

- COMPASS application
  - Uses LSM303C eCompass MEMS device
    - 3D accelerometer and 3D magnetometer
  - Sub-Menu
    - “CALIB”: rotate board (360° on all axis's) after scrolling message invitation
    - “RUN”: displays angle in degrees
  - LEFT key to exit



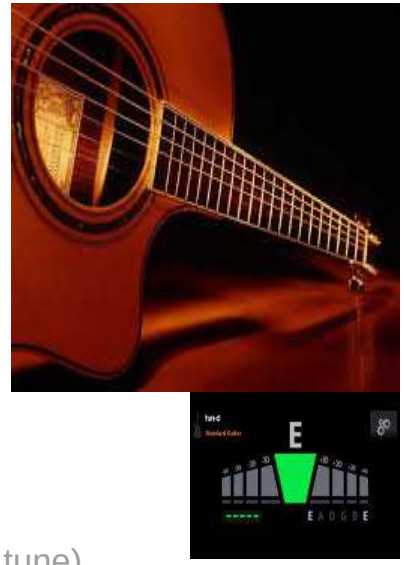


- SOUND meter application
  - Uses MP34DT01 audio sensor to measure ambient noise
  - Displays measurement value in dB on LCD screen
  - LEFT key to exit

life.augmented

Guitar tuner demonstration

- Select guitar string
  - “STR1”: “E” (low E, thickest string, closest to the ceiling)
  - “STR2”: “A
  - “STR3”: “D”
  - “STR4”: “G”
  - “STR5”: “B”
  - “STR6”: “e” (high E, thinnest string, closest to the floor)

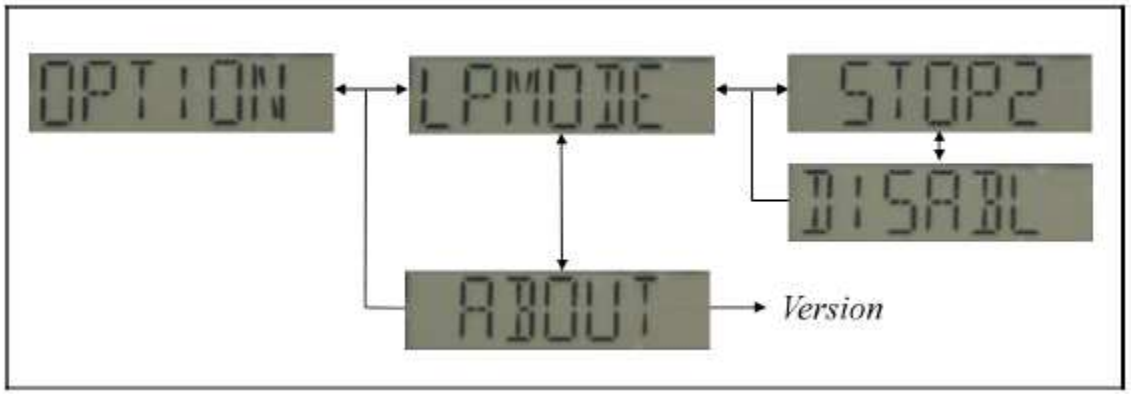


- RIGHT/SEL to start recording
- Output:
  - “ ++ “ when string needs to be tightened
  - “ + “ when string needs to be slightly tightened (close to correct tune)
  - “ OK “ when string is correctly tuned
  - “ - “ when string needs to be slightly loosened (close to correct tune)
  - “ -- ” when string needs to be loosened
- LEFT key to exit

life.augmented

Option app

- Select whether to enter STOP2 mode after few seconds of inactivity
- Display Firmware version #



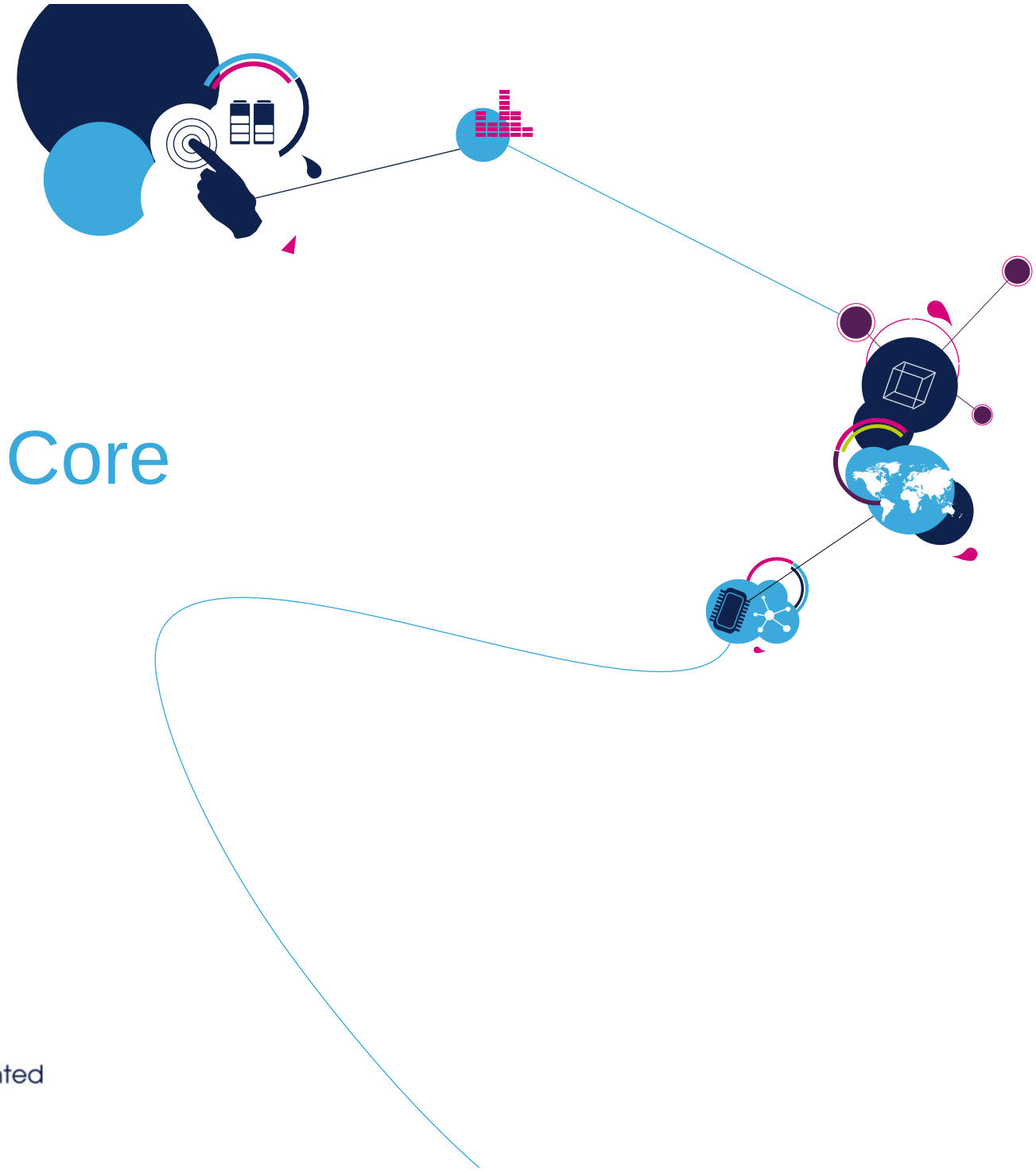
References

- Refer to www.st.com/stm32l4-discovery
  - Ordering information
  - Getting Started Manual, User's Manual and Application notes
  - Board Schematics
  - Application development environments support
  - Demonstration firmware sources

- Video available on YouTube and st.com
  - "Getting started with STM32L476 discovery kit for ultra-low-power & performance applications"

  - https://www.youtube.com/watch?v=UkTFORUS29Q

life.augmented



Cortex-M4 Core

life.augmented

Cortex-M processors

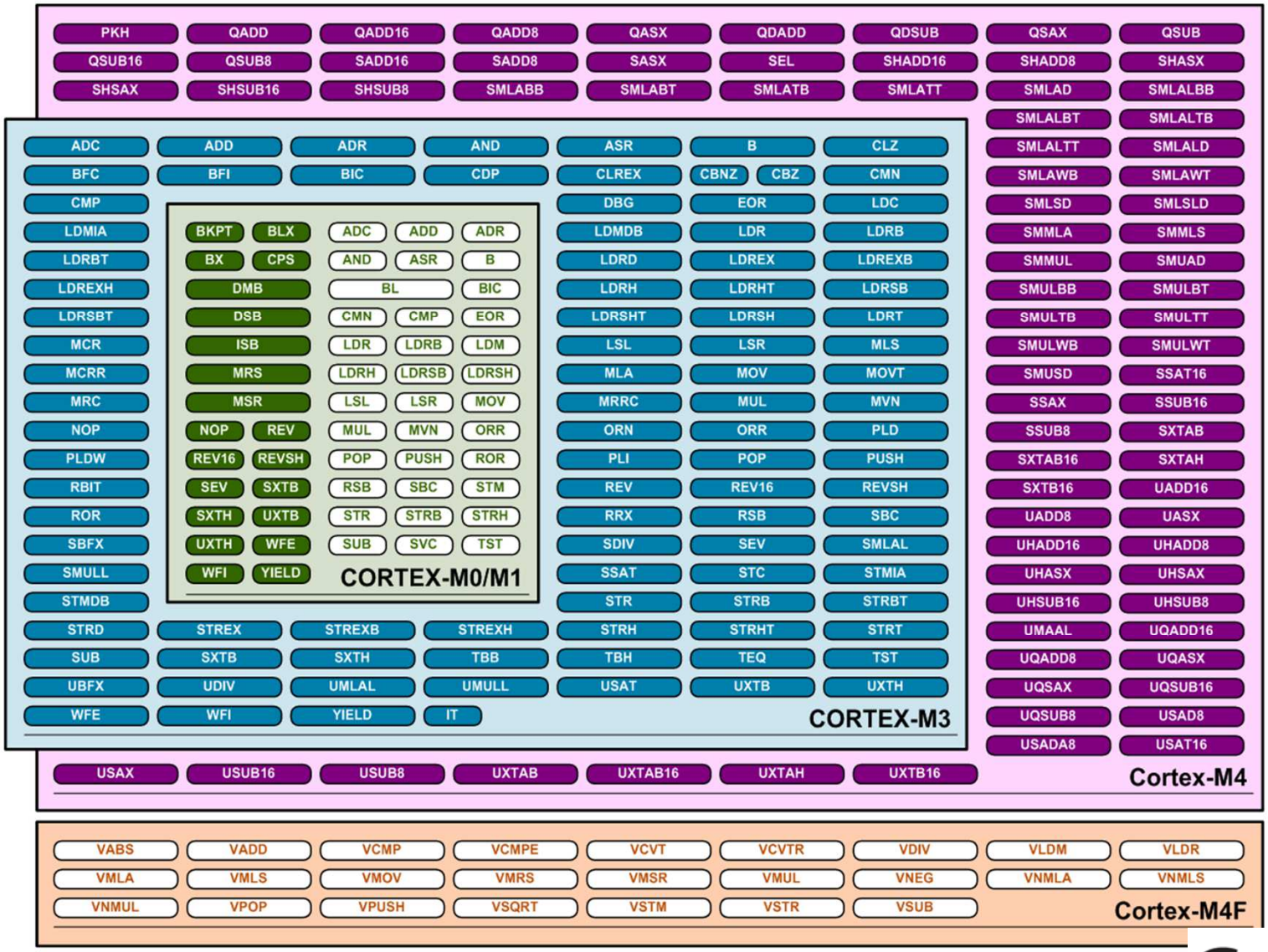
- **Forget traditional 8/16/32-bit classifications**
  - Seamless architecture across all applications
  - Every product optimised for ultra low power and ease of use

Cortex-M0	Cortex-M3	Cortex-M4	Cortex-M7
“8/16-bit” applications	**“16/32-bit” applications**	**“32-bit/DSC” apps**	**Hi-Performance apps**

Binary and tool compatible

life.augmented

Cortex-M processors binary compatible



life.augmented

Cortex
Low-Power Leadership from ARM

ARM Cortex M4 Core

ARM

Cortex
Low-Power Leadership from ARM

FPU

Single precision
Ease of use
Better code efficiency
Faster time to market
Eliminate scaling and saturation
Easier support for meta-language tools

What is Cortex-M4?

MCU

Ease of use of C programming
Interrupt handling
Ultra-low power

Cortex-M4

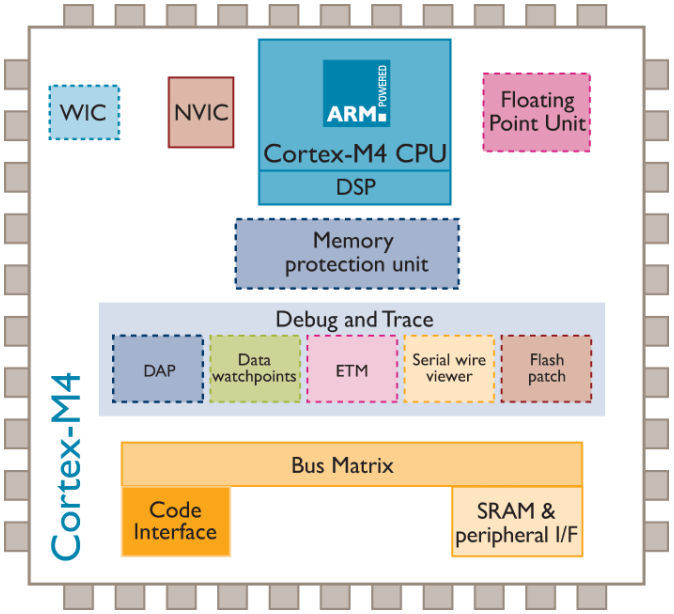
DSP

Harvard architecture
Single-cycle MAC
Barrel shifter

life.augmented

Cortex-M4 processor microarchitecure

- **ARMv7ME Architecture**
  - Thumb-2 Technology
  - DSP and SIMD extensions
  - Single cycle MAC (Up to 32 x 32 + 64 -> 64)
  - Optional single precision FPU
  - Integrated configurable NVIC
  - Compatible with Cortex-M3
- **Microarchitecture**
  - 3-stage pipeline with branch speculation
  - 3x AHB-Lite Bus Interfaces
- **Configurable for ultra low power**
  - Deep Sleep Mode, Wakeup Interrupt Controller
  - Power down features for Floating Point Unit
- **Flexible configurations for wider applicability**
  - Configurable Interrupt Controller (1-240 Interrupts and Priorities)
  - Optional Memory Protection Unit
  - Optional Debug & Trace



life.augmented

Cortex
Low-Power Leadership from ARM

Cortex-M4 extended single cycle MAC

OPERATION	INSTRUCTIONS	CM3	CM4
16 x 16 = 32	SMULBB, SMULBT, SMULTB, SMULTT	n/a	1
16 x 16 + 32 = 32	SMLABB, SMLABT, SMLATB, SMLATT	n/a	1
16 x 16 + 64 = 64	SMLALBB, SMLALBT, SMLALTB, SMLALTT	n/a	1
16 x 32 = 32	SMULWB, SMULWT	n/a	1
(16 x 32) + 32 = 32	SMLAWB, SMLAWT	n/a	1
(16 x 16) ± (16 x 16) = 32	SMUAD, SMUADX, SMUSD, SMUSDX	n/a	1
(16 x 16) ± (16 x 16) + 32 = 32	SMLAD, SMLADX, SMLSD, SMLSDX	n/a	1
(16 x 16) ± (16 x 16) + 64 = 64	SMLALD, SMLALDX, SMLSLD, SMLSLDX	n/a	1
32 x 32 = 32	MUL	1	1
32 ± (32 x 32) = 32	MLA, MLS	2	1
32 x 32 = 64	SMULL, UMULL	5-7	1
(32 x 32) + 64 = 64	SMLAL, UMLAL	5-7	1
(32 x 32) + 32 + 32 = 64	UMAAL	n/a	1
32 ± (32 x 32) = 32 (upper)	SMMLA, SMMLAR, SMMLS, SMMLSR	n/a	1
(32 x 32) = 32 (upper)	SMMUL, SMMULR	n/a	1

All the above operations are single cycle on the Cortex-M4 processor

life.augmented

Cortex-M4 DSP instructions compared

		Cycle counts	
CLASS	INSTRUCTION	CORTEX-M3	Cortex-M4
Arithmetic	ALU operation (not PC)	1	1
	ALU operation to PC	3	3
	CLZ	1	1
	QADD, QDADD, QSUB, QDSUB	n/a	1
	QADD8, QADD16, QSUB8, QSUB16	n/a	1
	QDADD, QDSUB	n/a	1
	QASX, QSAX, SASX, SSAX	n/a	1
	SHASX, SHSAX, UHASX, UHSAX	n/a	1
	SADD8, SADD16, SSUB8, SSUB16	n/a	1
	SHADD8, SHADD16, SHSUB8, SHSUB16	n/a	1
	UQADD8, UQADD16, UQSUB8, UQSUB16	n/a	1
	UHADD8, UHADD16, UHSUB8, UHSUB16	n/a	1
	UADD8, UADD16, USUB8, USUB16	n/a	1
	UQASX, UQSAX, USAX, UASX	n/a	1
	UXTAB, UXTAB16, UXTAH	n/a	1
	USAD8, USADA8	n/a	1
Multiplication	MUL, MLA	1 - 2	1
	MULS, MLAS	1 - 2	1
	SMULL, UMULL, SMLAL, UMLAL	5 - 7	1
	SMULBB, SMULBT, SMULTB, SMULTT	n/a	1
	SMLABB, SMLBT, SMLATB, SMLATT	n/a	1
	SMULWB, SMULWT, SMLAWB, SMLAWT	n/a	1
	SMLALBB, SMLALBT, SMLALTB, SMLALTT	n/a	1
	SMLAD, SMLADX, SMLALD, SMLALDX	n/a	1
	SMLSD, SMLSDX	n/a	1
	SMLSLD, SMLSLD	n/a	1
	SMMLA, SMMLAR, SMMLS, SMMLSR	n/a	1
	SMMUL, SMMULR	n/a	1
	SMUAD, SMUADX, SMUSD, SMUSDX	n/a	1
	UMAAL	n/a	1
Division	SDIV, UDIV	2 - 12	2 – 12

Single cycle MAC

life.augmented

FPU arithmetic instructions

Operation	Description	Assembler	Cycle
Absolute value	of float	VABS.F32	1
Negate	float and multiply float	VNEG.F32 VNMUL.F32	1 1
Addition	floating point	VADD.F32	1
Subtract	float	VSUB.F32	1
Multiply	float then accumulate float then subtract float then accumulate then negate float the subtract the negate float	VMUL.F32 VMLA.F32 VMLS.F32 VNMLA.F32 VNMLS.F32	1 3 3 3 3
Multiply (fused)	then accumulate float then subtract float then accumulate then negate float then subtract then negate float	VFMA.F32 VFMS.F32 VFNMA.F32 VFNMS.F32	3 3 3 3
Divide	float	VDIV.F32	14
Square-root	of float	VSQRT.F32	14

DSP lib provided for free by ARM

- The benefits of software libraries for Cortex-M4
  - Enables end user to develop applications faster
    - Keeps end user abstracted from low level programming
  - Benchmarking vehicle during system development
  - Clear competitive positioning against incumbent DSP/DSC offerings
  - Accelerate third party software development
- Keeping it easy to access for end user
  - Minimal entry barrier - very easy to access and use
- One standard library – no duplicated efforts
  - ARM channels effort/resources with software partner
  - Value add through another level of software – eg: filter config tools

life.augmented

Cortex
Low-Power Leadership from ARM

DSP lib function list snapshot

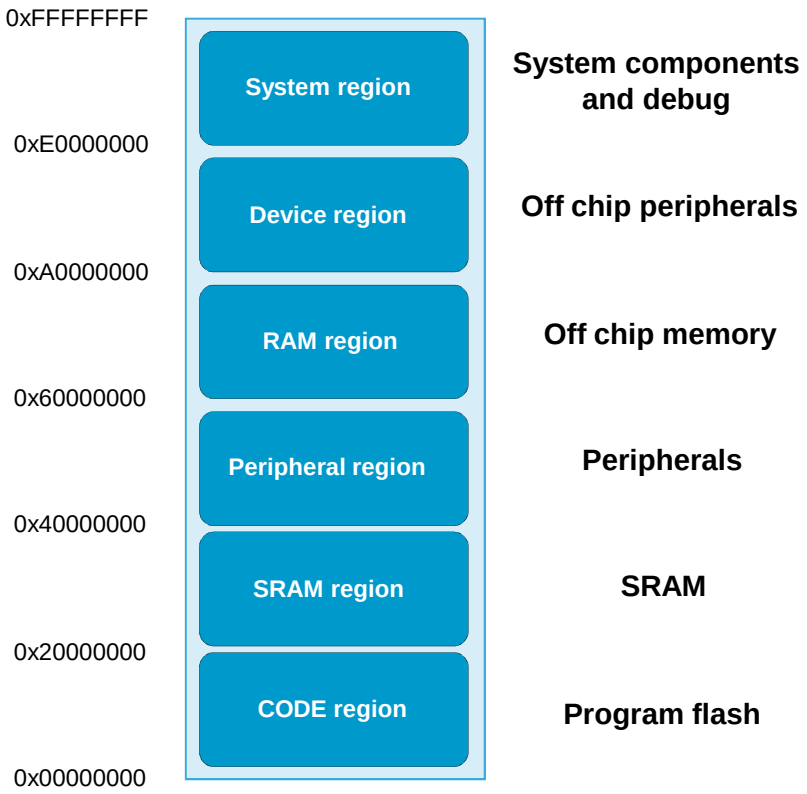
- Basic math – vector mathematics
- Fast math – sin, cos, sqrt etc
- Interpolation – linear, bilinear
- Complex math
- Statistics – max, min,RMS etc
- Filtering – IIR, FIR, LMS etc
- Transforms – FFT(real and complex) , Cosine transform etc
- Matrix functions
- PID Controller
- Support functions – copy/fill arrays, data type conversions etc

life.augmented

Cortex
Low-Power Leadership from ARM

Memory map overview

- **4GB linear memory space**
- **No paging/banking**
- **All locations always accessible by the SW**
- **Supports 8/16/32-bit data**



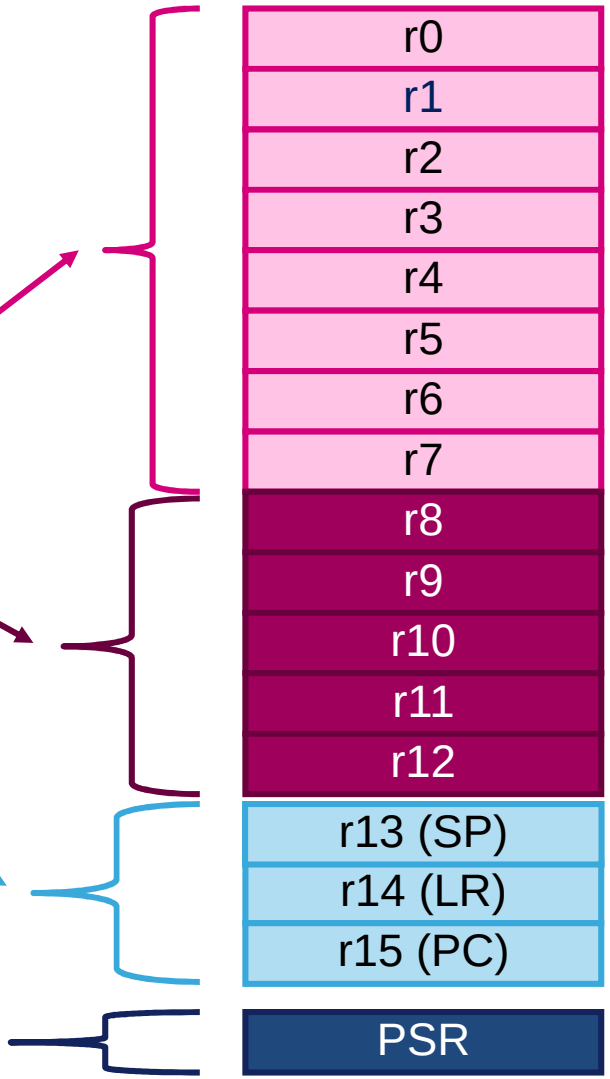
- **Standard across all Cortex-M implementations**

life.augmented

Core Registers

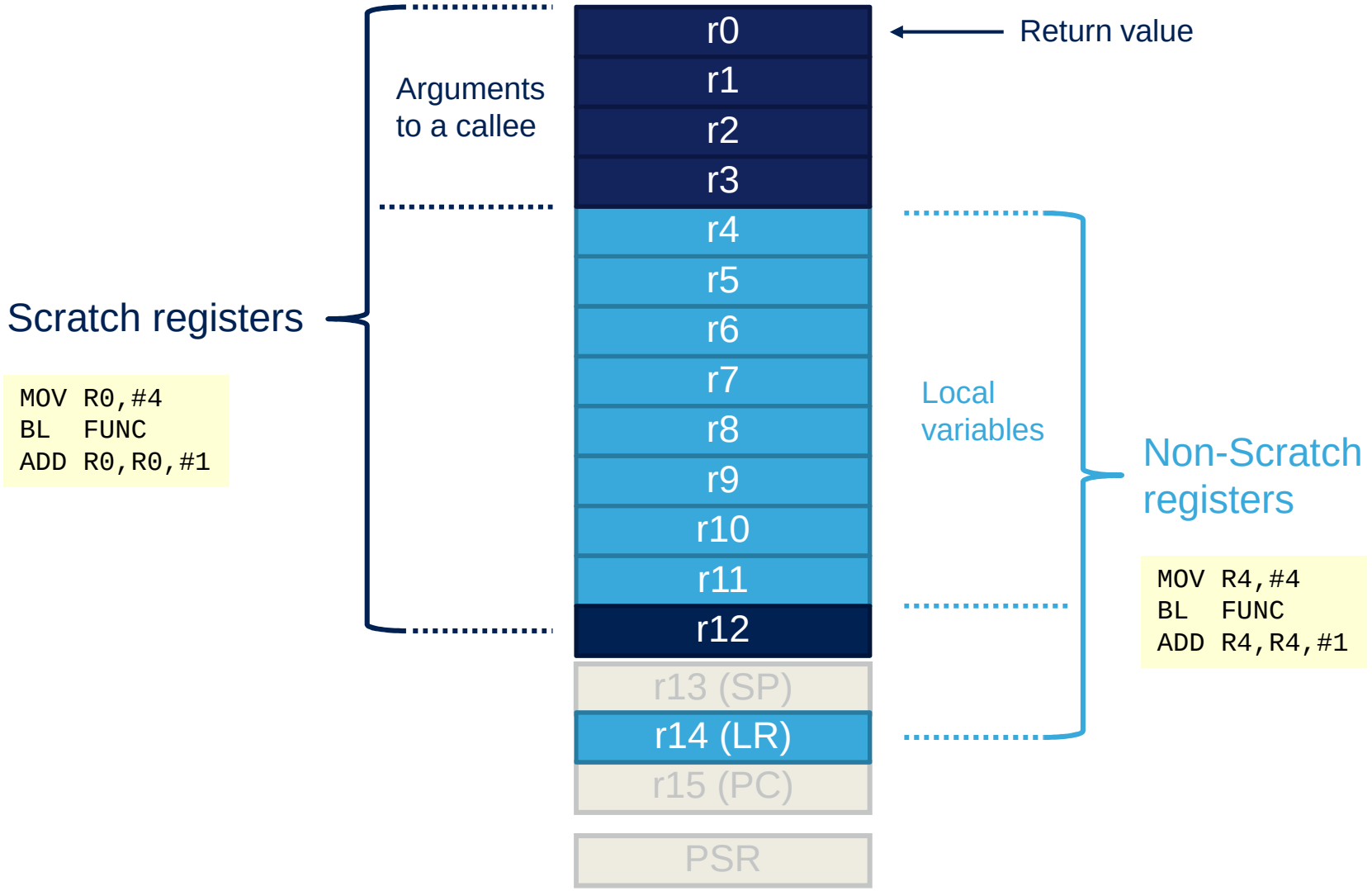
- **All registers are 32-bit wide**
- Instructions exist to efficiently support packed 8/16/32-bit data in memory
- **13 general purpose registers**
  - Registers r0 – r7 (**Low registers**)
  - Registers r8 – r12 (**High registers**)
- **Only 3 special registers** (MRS, MSR)
  - Stack Pointer (SP) – r13
  - Link Register (LR) – r14
  - Program Counter (PC) – r15
- **Program Status Register**

  (Application / Interrupt / Execution)



life.augmented

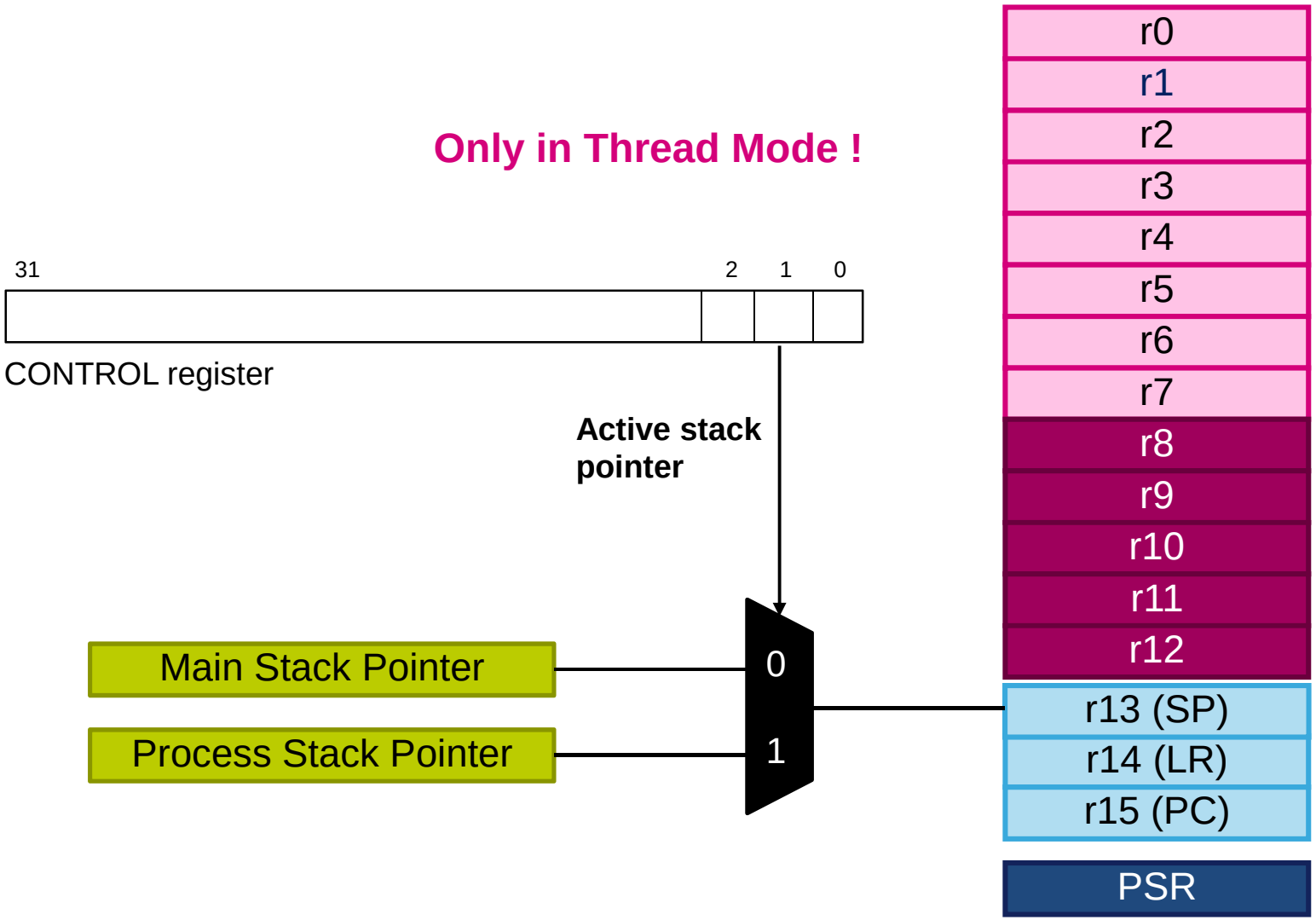
Register Usage Convention



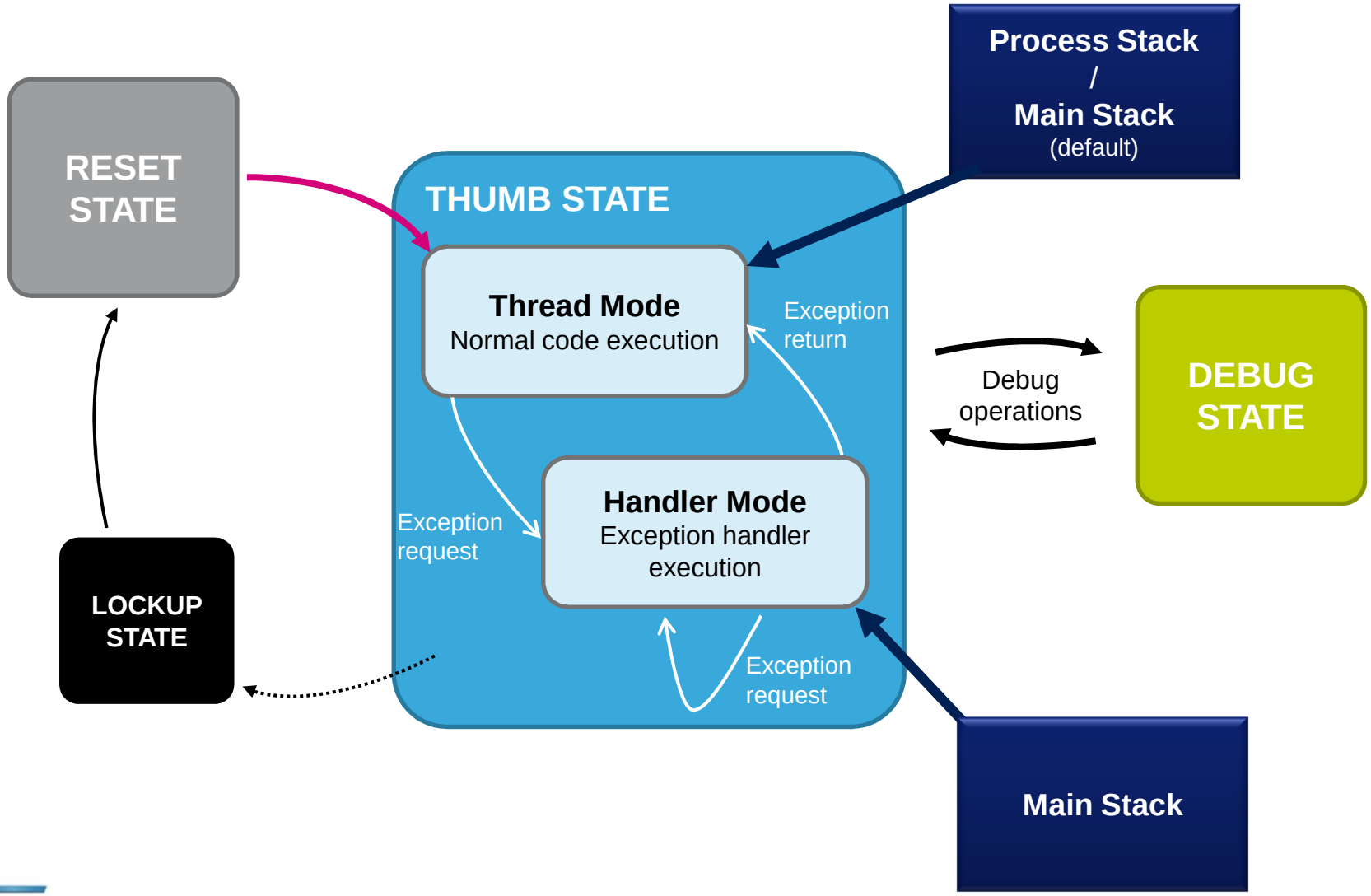
- ARM Architecture Procedure Call Standard (AAPCS)

Stack Pointer and Stacks

Only in Thread Mode !



Processor modes



Exception Model – Types and Priorities

No.	Exception Type	Priority	Type of Priority	Descriptions
1	Reset	-3 (Highest)	fixed	Reset
2	NMI	-2	fixed	Non-Maskable Interrupt
3	**Hard Fault**	**-1**	**fixed**	**Default fault if other hander not implemented**
4-10	Reserved	N.A.	N.A.	
11	**SVCall**	**Programmable**	**settable**	**Supervisor call (SVC)**
12-13	Reserved	N.A.	N.A.	
14	**PendSV**	**Programmable**	**settable**	**Request for System-Level Service**
15	SYSTICK	Programmable	settable	System Tick Timer
16	Interrupt #0	Programmable	settable	External Interrupt #0
……	……..	……..	settable	……..
97	Interrupt#81	Programmable	settable	External Interrupt #31

4 priority levels

life.augmented

Interrupt Entry and Return

- Interrupt handling is micro-coded

  → **No instruction overhead**

- Entry (“**Stacking**”)
  - Processor state automatically saved to the stack over the bus
    - **{PC, xPSR, R0-R3, R12, LR}**
    - **What about the other registers?**
  - Then, ISR ready to start executing as soon as stack **PUSH** complete
- Exit (“**Unstacking**”)
  - Processor state is automatically restored from the stack
  - Then interrupted instruction is executed upon completion of stack **POP**

life.augmented

Preemption

- *“Interruption of the exception handler”*
- **Depending on the priority**
- If higher priority exception request comes while another lower priority exception handler is executing

  → the higher priority exception can **preempt** the lower priority exception handler

- Such exceptions are called **nested exceptions**

Exception Response

- Standard exception latency is **12-cycles**
  - The latency from processor clock cycle time when the exception is asserted to the first instruction of the exception handler execution
  - **0 memory wait states**
- Can be reduced by the following two mechanisms:
  - **Tail Chaining**

    If exception request exists or occurs just as another exception handler returns

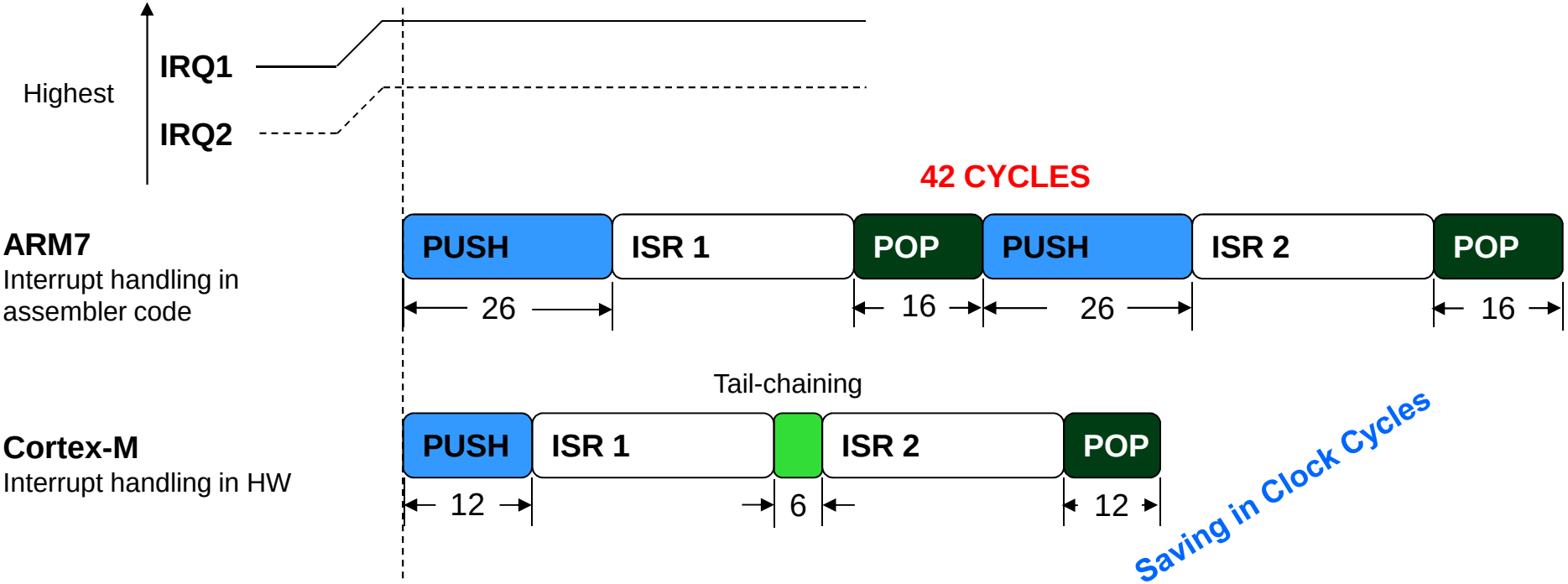
    → Stacking process can be skipped
  - **Late arriving** (Late arrival)

    If higher priority exception request occurs during lower prority exception stacking process

    → Higher priority exception handler is called right after the stacking finishes

life.augmented

Interrupt Response – Tail-Chaining



ARM7

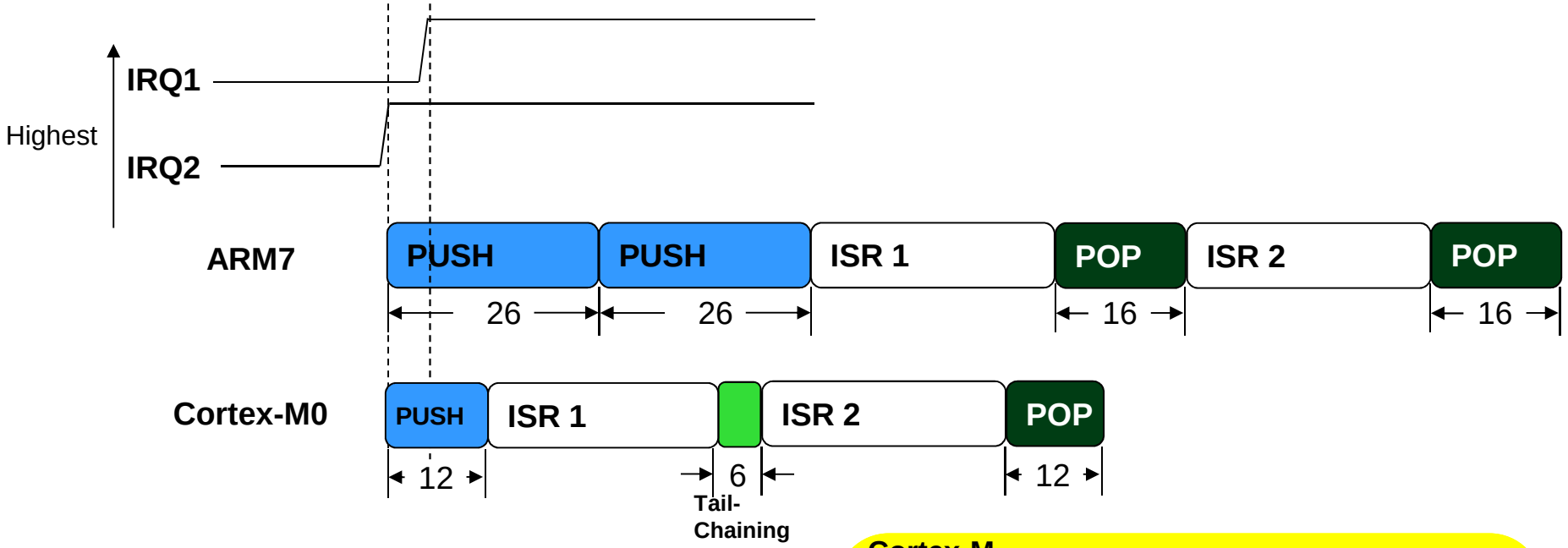
- 26 cycles from IRQ1 to ISR1 entered
  - Up to 42 cycles if LSM
- 42 cycles from ISR1 exit to ISR2 entry
- 16 cycles to return from ISR2

Cortex-M

- 12 cycles from IRQ1 to ISR1 entered
- 6 cycles from ISR1 exit to ISR2 entry
- 12 cycles to return from ISR2

life.augmented

Interrupt Response – Late Arriving



ARM7

- 26 cycles to ISR2 entered
- Immediately pre-empted by IRQ1 and takes a further 26 cycles to enter ISR 1.
- ISR 1 completes and then takes 16 cycles to return to ISR 2.

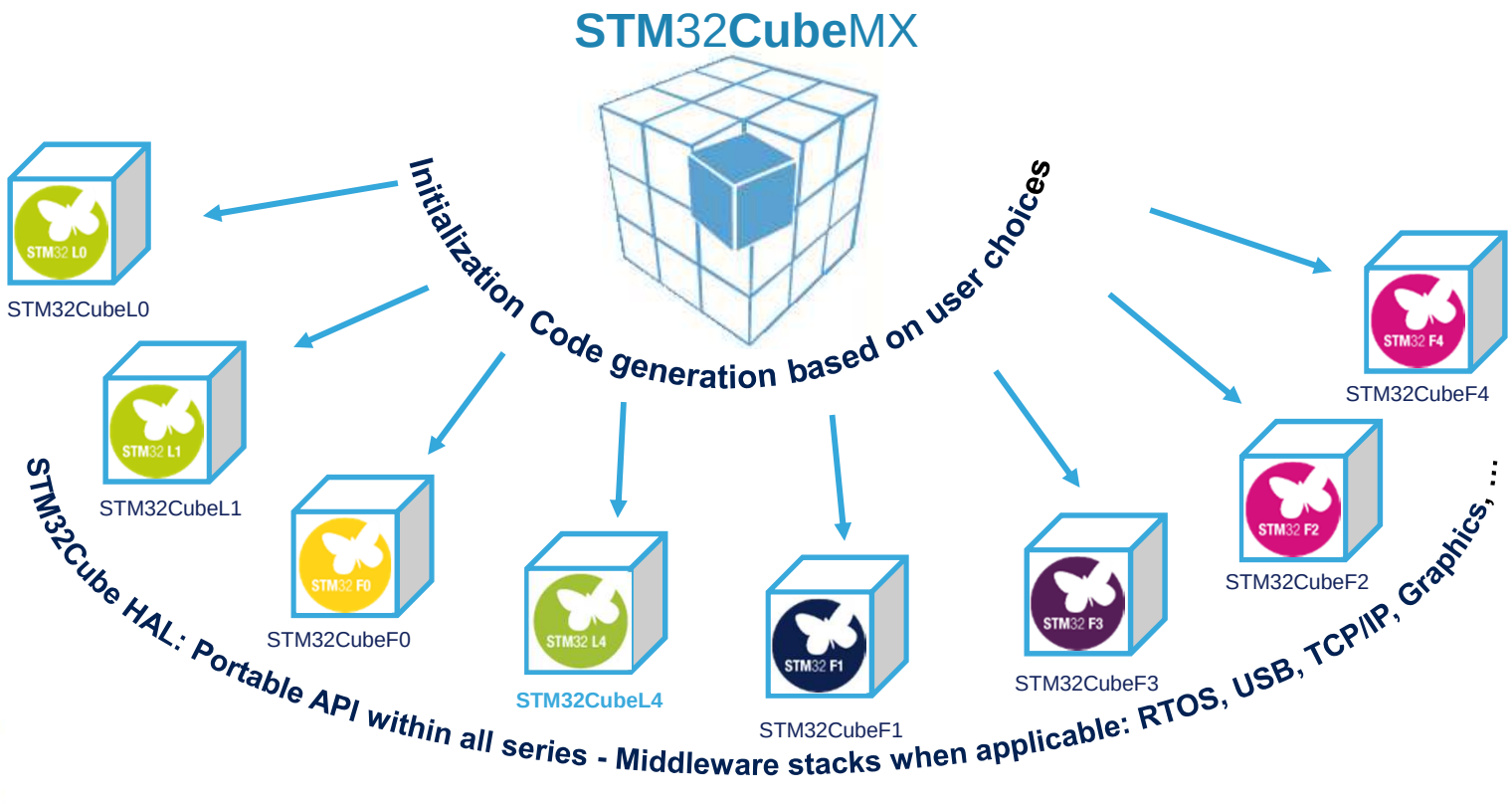
Cortex-M

- Stack push to ISR 2 is interrupted
- Stacking continues but new vector address is fetched in parallel
- Late-arrival to ISR1 entry will depend of the PUSH status, then 4 cycles will be necessary to read the vector table.
- Tail-chain into ISR 2

life.augmented

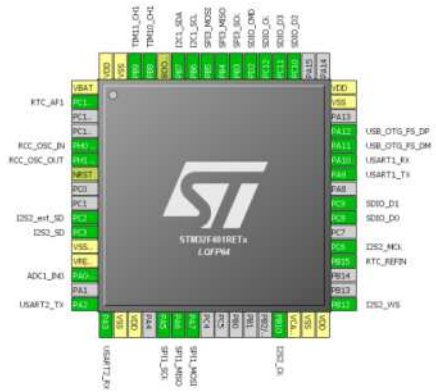
STM32Cube™ Introduction

- STM32Cube™ includes:
  - A configuration tool, STM32CubeMX generating initialization code from user choices
  - Firmware offering, delivered per series (like STM32CubeF4) with:
    - An STM32 Abstraction Layer embedded software: STM32Cube HAL
    - A consistent set of Middleware: RTOS, USB, TCP/IP, Graphics, …



life.augmented

STM32CubeMX



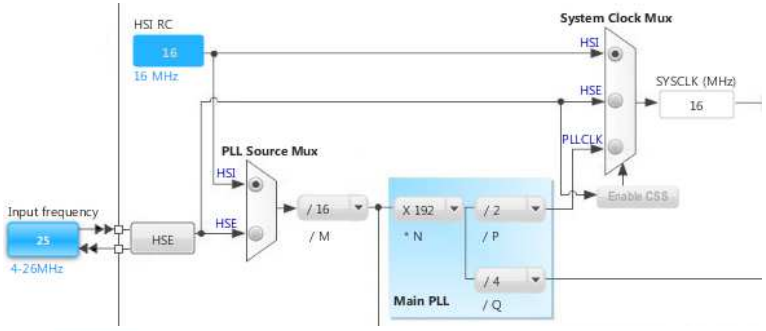
Pinout Wizard

Basic Parameters	
Baud Rate	115200 Bits/s
Word Length	8 Bits (including Parity)
Parity	None
Stop Bits	1
Advanced Parameters	
Data Direction	Receive and Transmit
Over Sampling	16 Samples

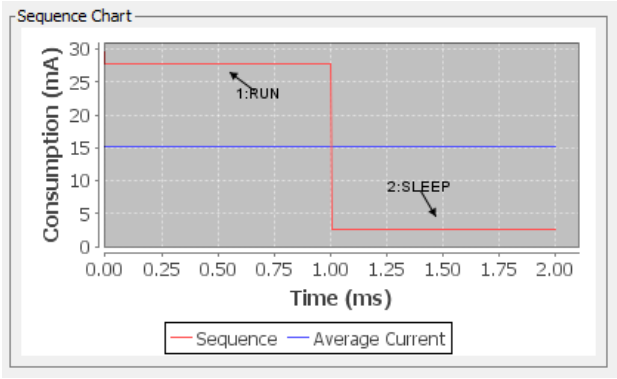
Baud Rate
BaudRate must be between **110 Bits/s** and **10.5 MBits/s**.

Peripherals & Middleware Wizard

Clock Tree wizard



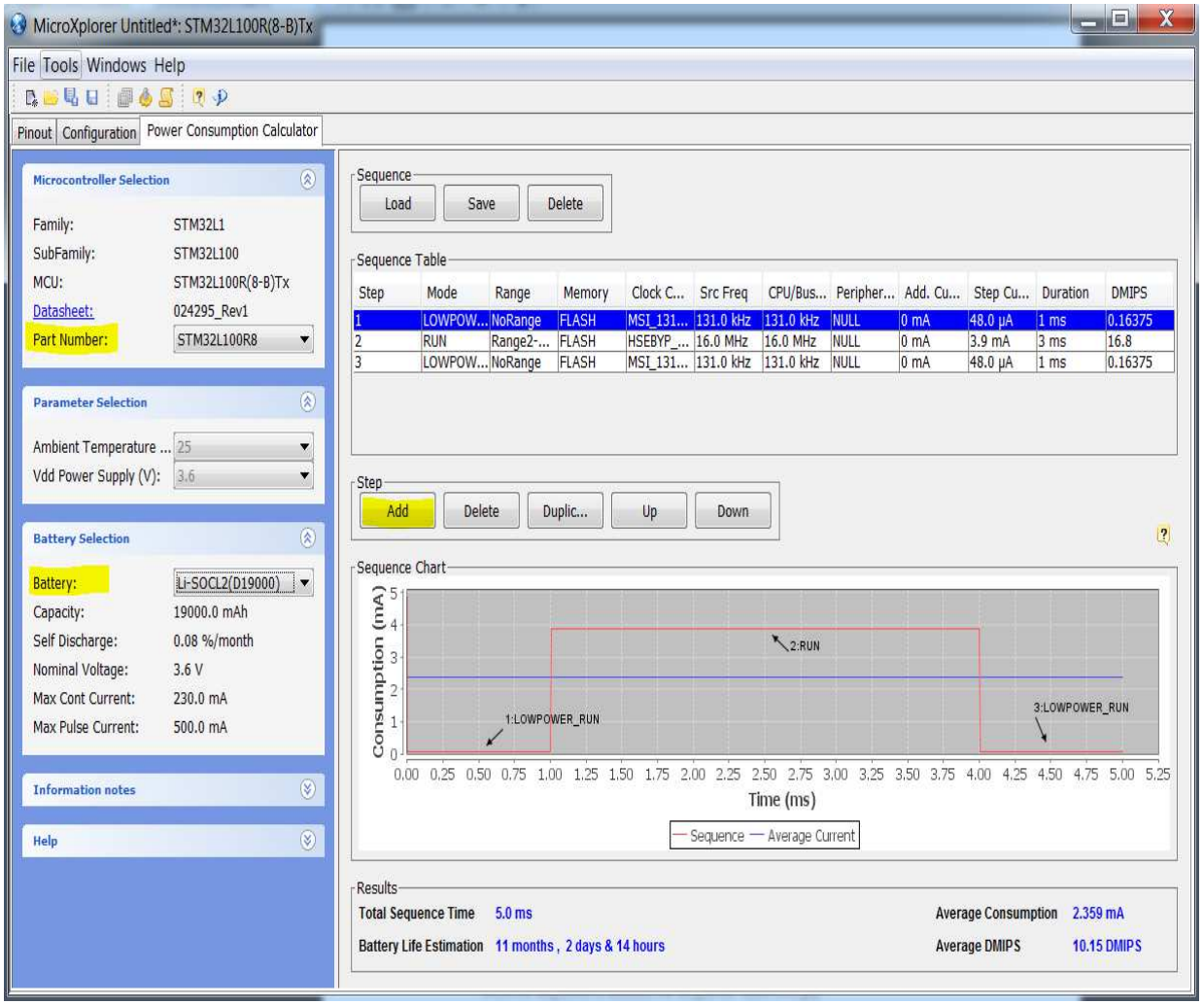
Power Consumption Wizard



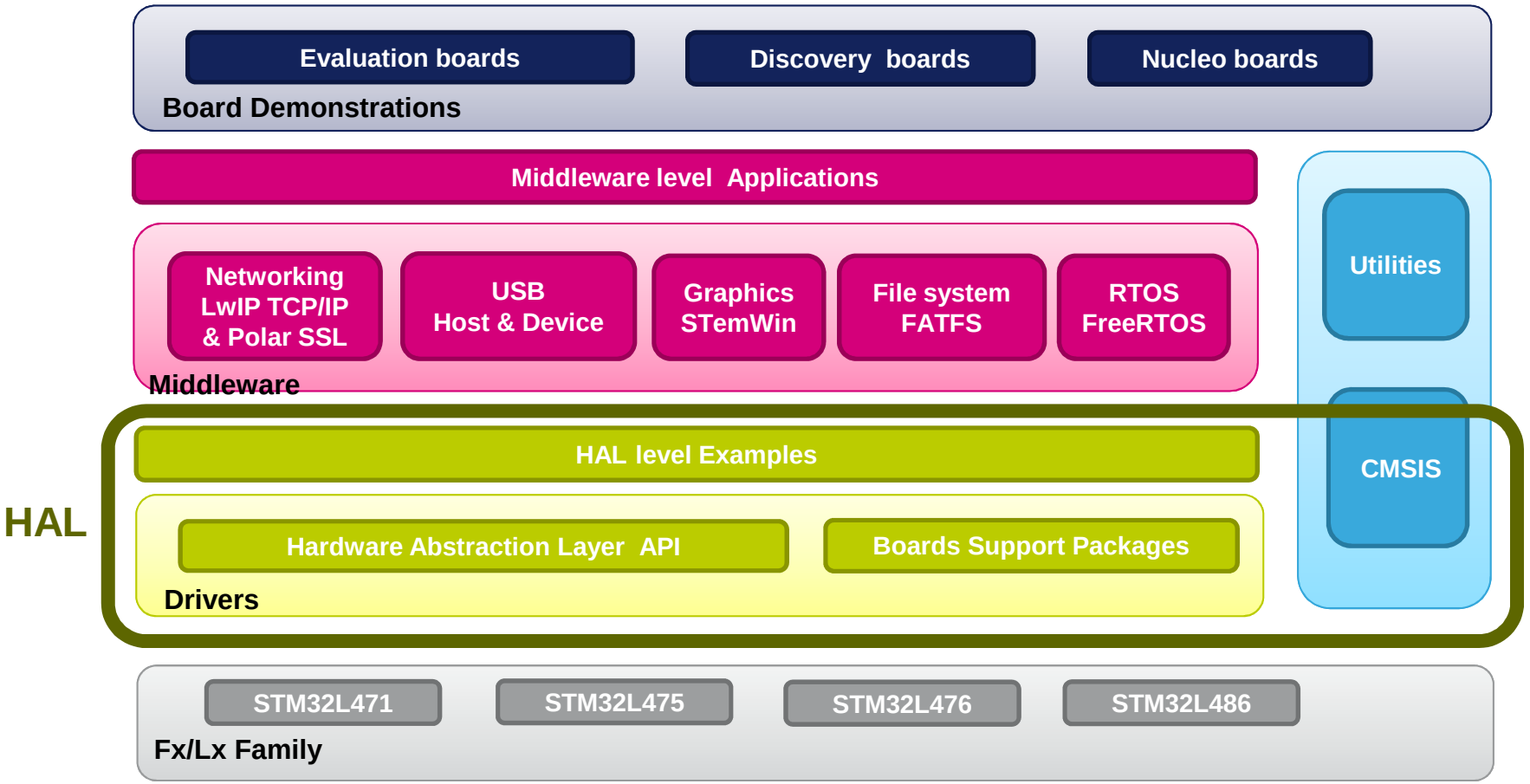
life.augmented

Power consumption calculator

- Power step definitions
- Battery selection
- Creation of consumption graph
- Display of
  - Average consumption
  - Average DMIPS
  - Battery lifetime


MicroXplorer Untitled*: STM32L100R(8-B)Tx
File Tools Windows Help
Pinout Configuration Power Consumption Calculator
Microcontroller Selection
Family: STM32L1
SubFamily: STM32L100
MCU: STM32L100R(8-B)Tx
Datasheet: 024295_Rev1
Part Number: STM32L100R8
Parameter Selection
Ambient Temperature ... 25
Vdd Power Supply (V): 3.6
Battery Selection
Battery: Li-SOCL2(D19000)
Capacity: 19000.0 mAh
Self Discharge: 0.08 %/month
Nominal Voltage: 3.6 V
Max Cont Current: 230.0 mA
Max Pulse Current: 500.0 mA
Information notes
Help
Sequence
Load Save Delete
Sequence Table
Step Mode Range Memory Clock C... Src Freq CPU/Bus... Peripher... Add. Cu... Step Cu... Duration DMIPS
1 LOWPOW... NoRange FLASH MSI_131... 131.0 kHz 131.0 kHz NULL 0 mA 48.0 µA 1 ms 0.16375
2 RUN Range2-... FLASH HSEBYP_... 16.0 MHz 16.0 MHz NULL 0 mA 3.9 mA 3 ms 16.8
3 LOWPOW... NoRange FLASH MSI_131... 131.0 kHz 131.0 kHz NULL 0 mA 48.0 µA 1 ms 0.16375
Step
Add Delete Duplic... Up Down
Sequence Chart
Consumption (mA)
Time (ms)
1:LOWPOWER_RUN
2:RUN
3:LOWPOWER_RUN
Sequence Average Current
Results
Total Sequence Time 5.0 ms
Average Consumption 2.359 mA
Battery Life Estimation 11 months , 2 days & 14 hours
Average DMIPS 10.15 DMIPS

STM32Cube Firmware Components

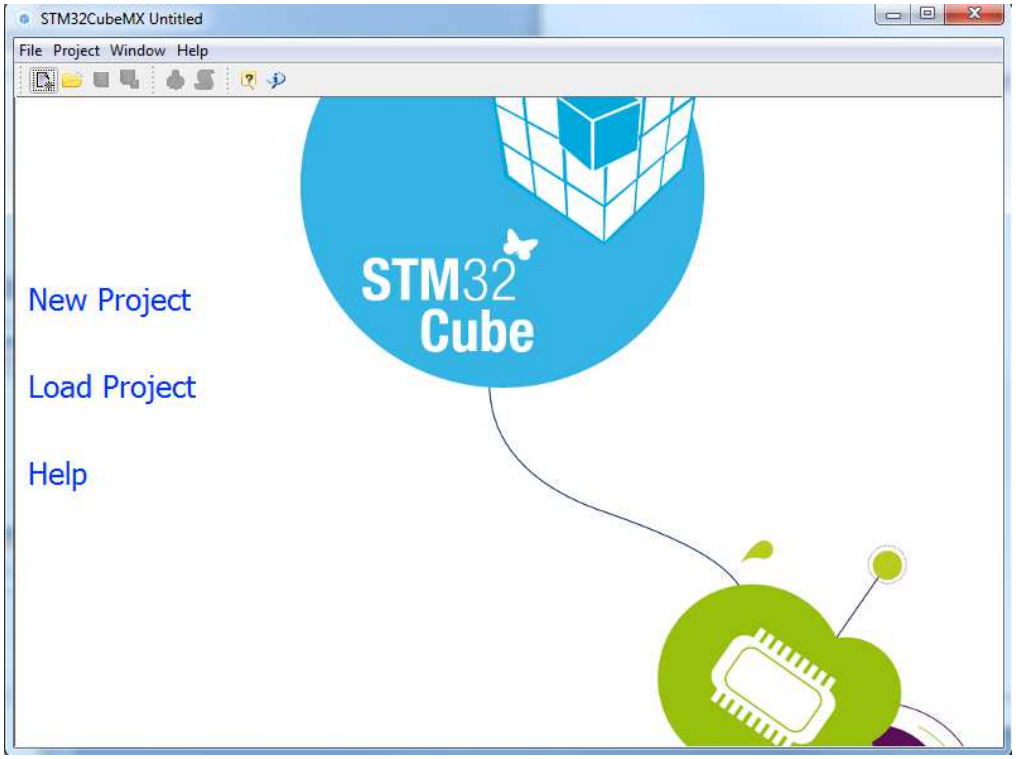


life.augmented

Hands-On Lab #1:
Getting Started with CubeMX
LED Blinky in Five Easy Steps!

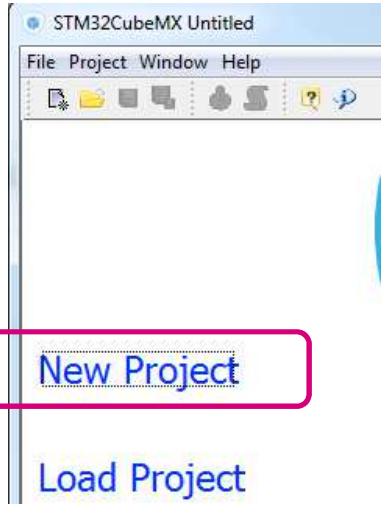
life.augmented

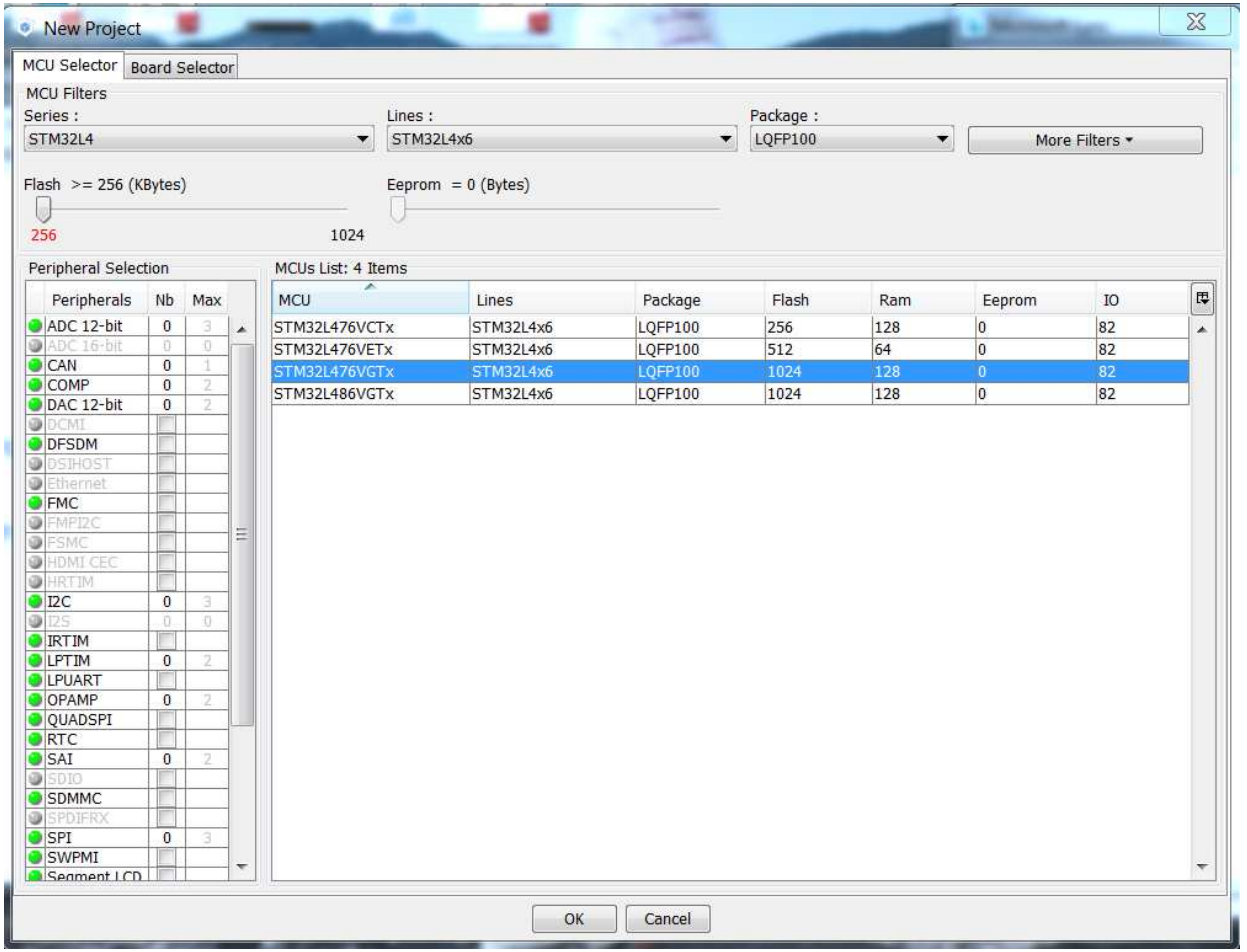
Run STM32CubeMX



Step 1: Create New Project

- Create **New Project**
- Select **STM32L476VGTx**
  - **LQFP100, 1024KB Flash**
- Click **“OK”**

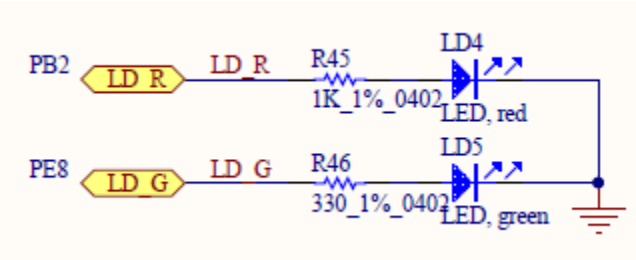


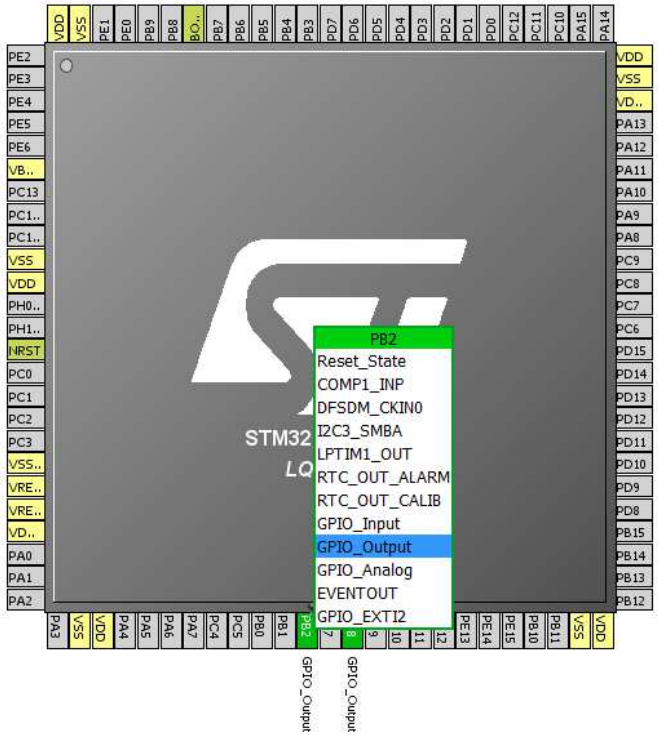


life.augmented

Step 2: Pin Configuration

- In this example we are going to use the LED's present on the STM32L476 Discovery board.
- Left-click PB2 & PE8 and set to **GPIO_Output** mode

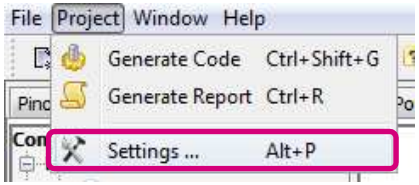




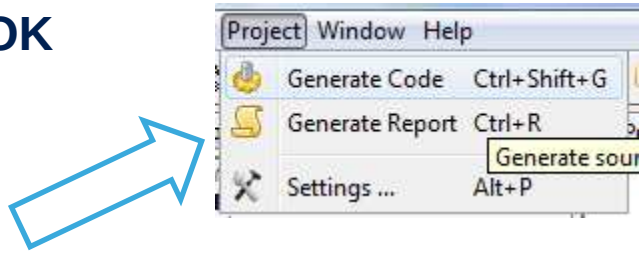
life.augmented

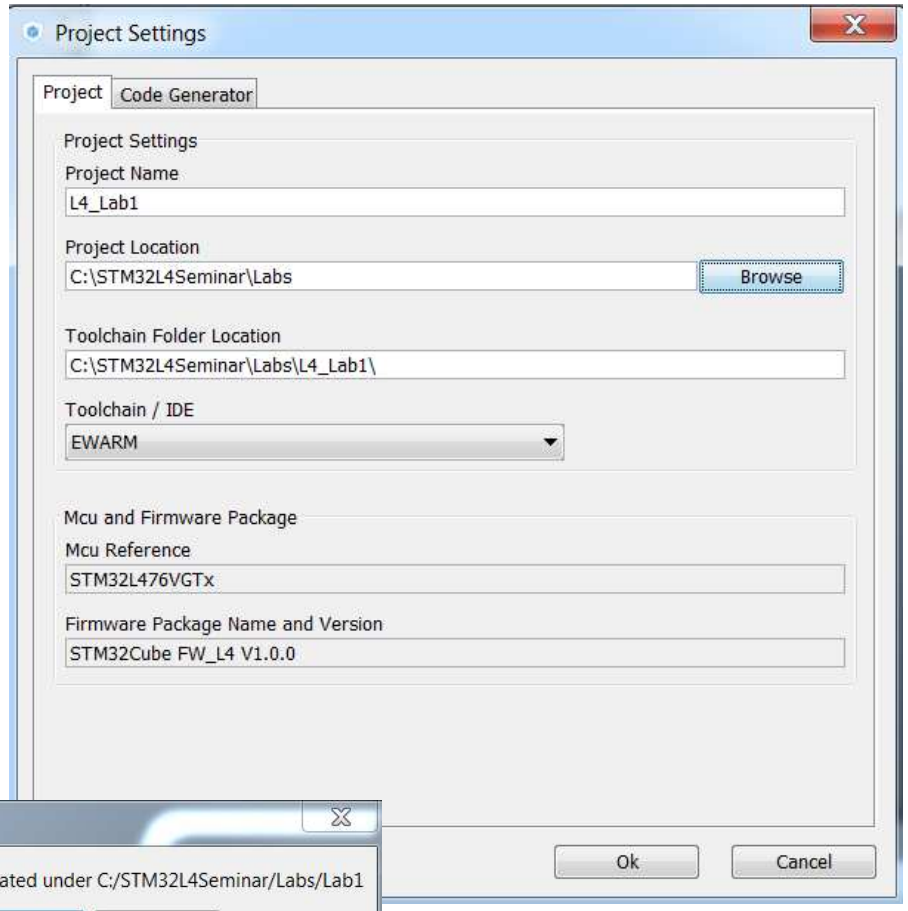
Step 3: Generate Source Code

- Open Project > **Settings** (Alt + P)

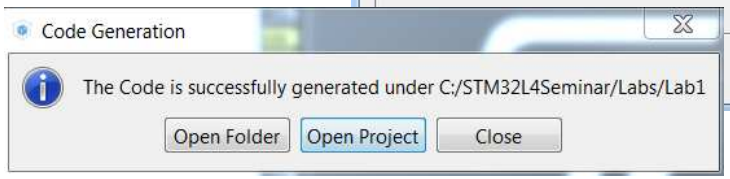


- Set the project name (Lab1) and the project location **(C:\STM32L4Seminar\Labs)**
- Set the IDE Toolchain to **EWARM**
- Click **OK**





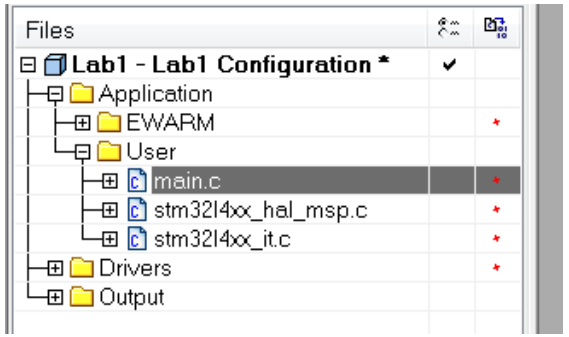
- **Generate Code** (Ctrl + Shift + G)
- Click **Open Project**



life.augmented

Step 4: Toggle The LED

- The IAR EWARM IDE should now be open.
- Expand the file tree and open the **main.c** file
- Add the following code inside the **while(1)** loop
  - Line 85 in “main.c”
  - Add within “USER CODE BEGIN WHILE” / “USER CODE END WHILE” section (this will preserve your code after regeneration)



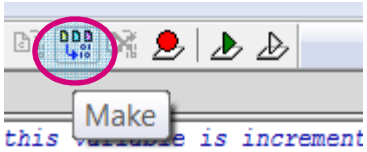

```
HAL_GPIO_TogglePin(GPIOB, GPIO_PIN_2);
HAL_Delay(100);
HAL_GPIO_TogglePin(GPIOE, GPIO_PIN_8);
HAL_Delay(100);
```

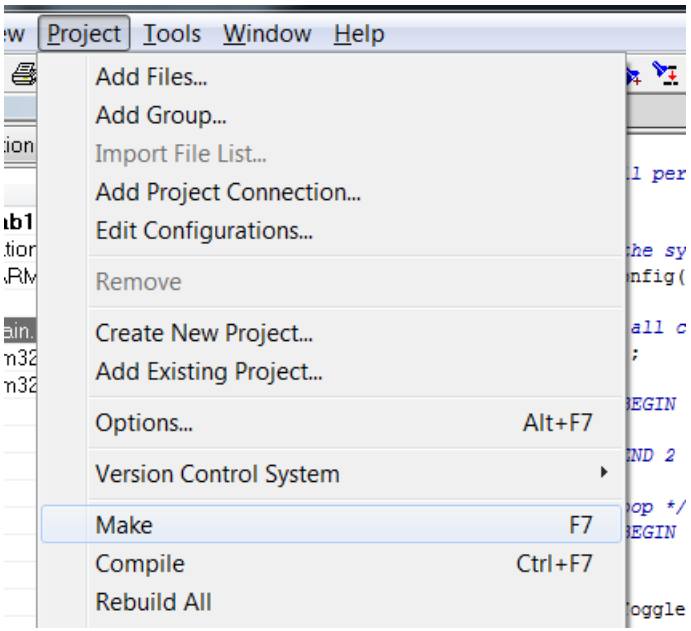
```
81    /* Infinite loop */
82    /* USER CODE BEGIN WHILE */
83    while (1)
84    {
85        HAL_GPIO_TogglePin(GPIOB, GPIO_PIN_2);
86        HAL_Delay(100);
87        HAL_GPIO_TogglePin(GPIOE, GPIO_PIN_8);
88        HAL_Delay(100);
89
90    /* USER CODE END WHILE */
91
```

life.augmented

Step 5: Build the Project

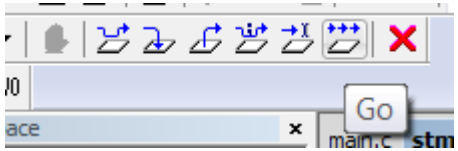
- Click “F7” or the “Make” button or use menu Project > Make.





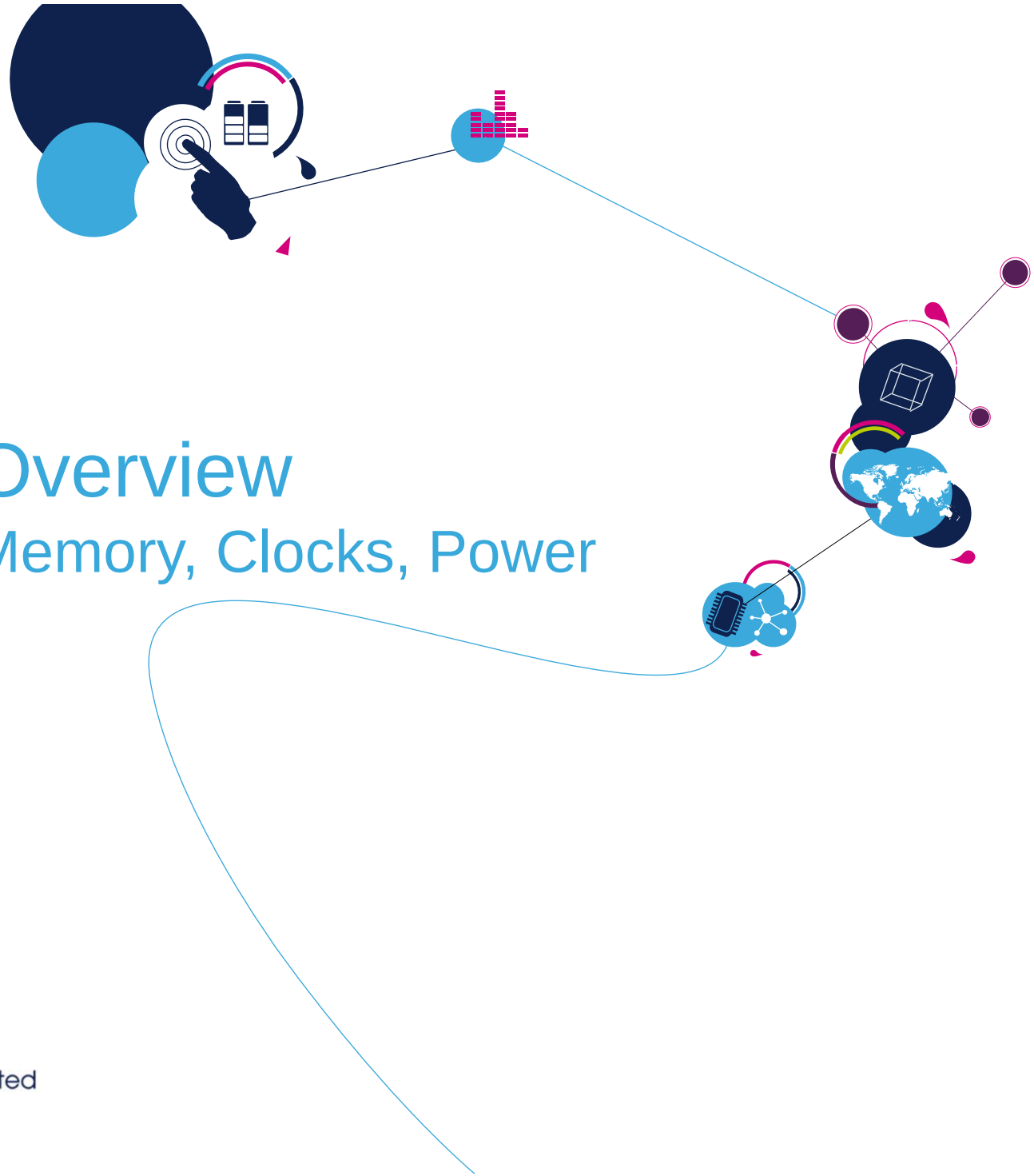
- Click the “Download and Debug” Green Arrow button (CTRL + D)

- Click the “Go” button (F5)



- Enjoy the flashing LED’s!

life.augmented



STM32L4 Overview

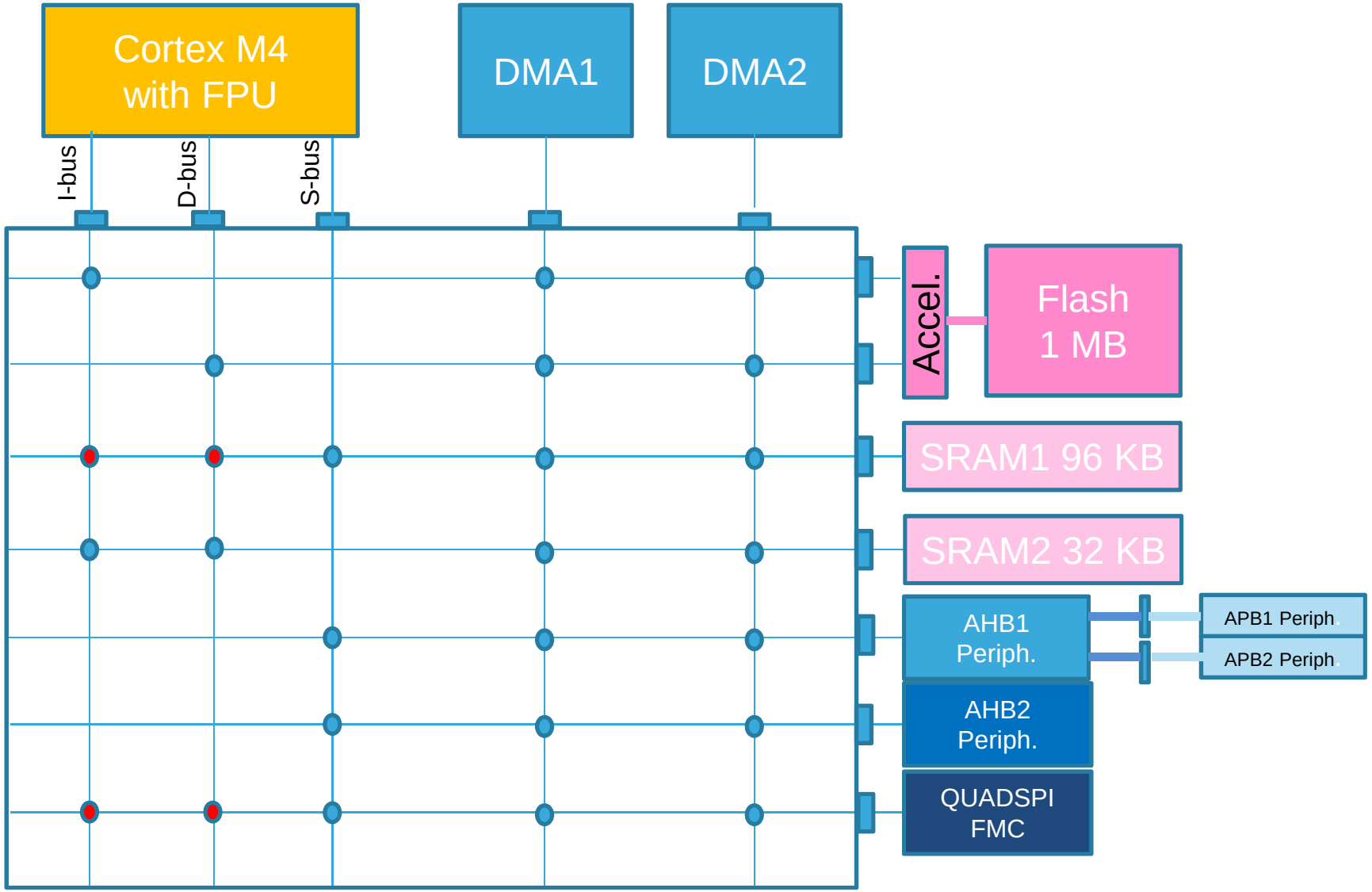
Architecture, Memory, Clocks, Power

life.augmented

STM32L476 block diagram

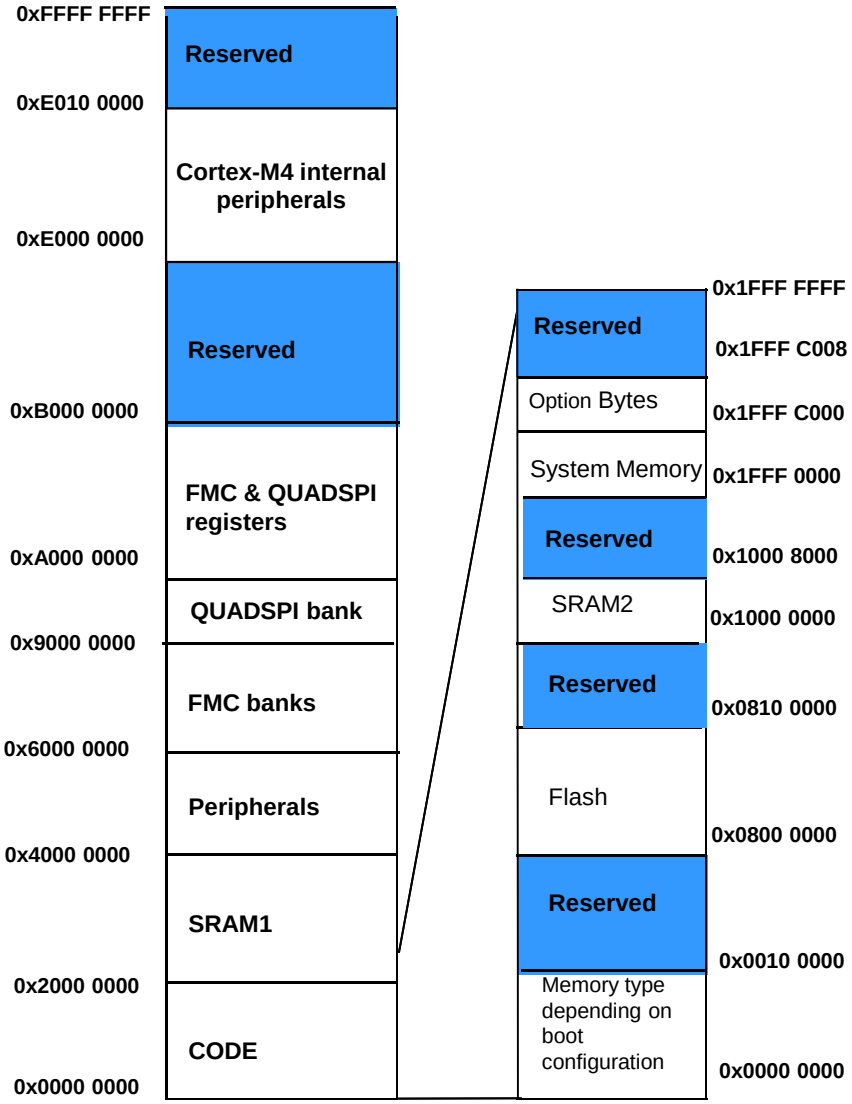
life.augmented

STM32L4 Bus matrix



Note: QuadSPI, FMC and SRAM1 I-bus & D-bus interfaces when remapped to *0x0000 0000* only

Memory Mapping



- **FLASH : up to 1 Mbytes, dual bank**
  - **FB_MODE = 0 in SYSCFG_MEMRMP:**
    - **Bank 1 @ 0x0800 0000**
    - **Bank 2 @ 0x0808 0000**
  - **FB_MODE = 1 in SYSCFG_MEMRMP**
    - **Bank 2 @ 0x0800 0000**
    - **Bank 1 @ 0x0808 0000**
- **SRAM: Up to 128 Kbytes SRAM split in 2 parts :**
  - **SRAM1 : 96 KBytes @2000 0000**
  - **SRAM2 : 32 KBytes @1000 0000 :**
    - Access through D-code and I-code
- **Physical remap at 0x0000 0000 selected by MEM_MODE in SYSCFG_MEMRMP:**
  - **Flash Bank 1 or Bank 2 (see FB_MODE)**
  - **System flash (bootloader)**
  - **FMC bank 1**
  - **SRAM1**
  - **QUADSPI**

life.augmented

Boot modes

Boot mode selection		Boot mode	Aliasing
BOOT1 (opposite of nBOOT1 option bit)	**BOOT0 (pin)**		
x	0	User Flash	Main Flash memory is selected as boot space
1	1	System memory	System memory is selected as boot space
0	1	SRAM1	Embedded SRAM1 is selected as boot space

- Flash Bank1 boot: Option Bit BFB2 = 0
- Flash Bank2 boot: Option Bit BFB2 = 1
- Bootloader supports: USART1/2/3, I2C1/2/3, SPI1/2/3, USB DFU, CAN

life.augmented

32KB SRAM2 features

- Access through D-code and I-code:
  - Code execution max performance without remap
- **HW parity check** : 4 bit per word
  - Enabled with **SRAM2_PE** user option bit
  - NMI / Timer Break on parity check error
- Optional **retention** in Standby mode
- **Write protection** with 1 Kbyte granularity
- **Read protection** with RDP :
  - Erased when RDP changed from Level 1 to Level 0
- **Software reset** and optional **Hardware reset** when system reset
  - Erased when setting **SRAM2ER** in SYSCFG_SCSR SRAM2 control and status register
  - Erased with system reset via **SRAM2_RST** user option bit

life.augmented

Flash organization

- Two 512KB User Flash banks:
  - Each bank is 256 pages of 2KB
- Information block:
  - System memory boot loader
  - 1 KB (128 double word) OTP for user data
  - Option bytes for user configuration

Flash area		Flash memory address	Size	Name
Main memory	Bank 1	0x0800 0000 – 0x0800 07FF	2K	Page 0
		…		…
		0x0807 F800 – 0x0807 FFFF	2K	Page 255
	Bank 2	0x0808 0000 – 0x0808 07FF	2K	Page 256
		…		…
		0x080F F800 – 0x080F FFFF	2K	Page 511
Information block	Bank 1	0x1FFF 0000 – 0x1FFF 6FFF	28K	System memory
	Bank 2	0x1FFF 8000 – 0x1FFF EFFF	28K	
	Bank 1	0x1FFF 7000 – 0x1FFF 73FF	1K	OTP area
	Bank 1	0x1FFF 7800 – 0x1FFF 780F	16	Option bytes
	Bank 2	0x1FFF F800 – 0x1FFF F80F	16	

life.augmented

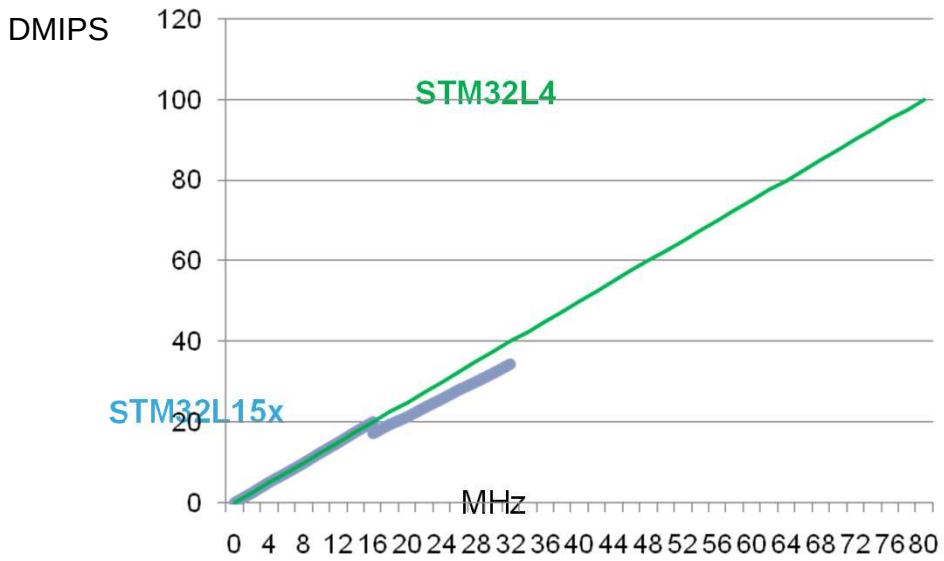
Program / Erase

- **ECC** (Error Code Correction) : 8-bit for 64-bit word
  - Single error correction:
    - Failure address and bank saved in FLASH_ECCR register, optional interrupt
  - Double error detection : NMI!
- Programming granularity is 64-bit
- Page granularity for erase is 2 Kbytes

Parameter	Typ
64-bit programming time	82 µs
Page (2 KB) erase time	22 ms
One row (32 double-word) programming time	Normal : 2.6 ms Fast : 1.9 ms
One page (2 KB) programming time	Normal : 20.9 ms Fast : 15.3 ms
One bank (512 KB) programming time	Normal : 5.35 s Fast : 3.9 s
Mass erase time (1 or 2 banks)	22 ms

Performances

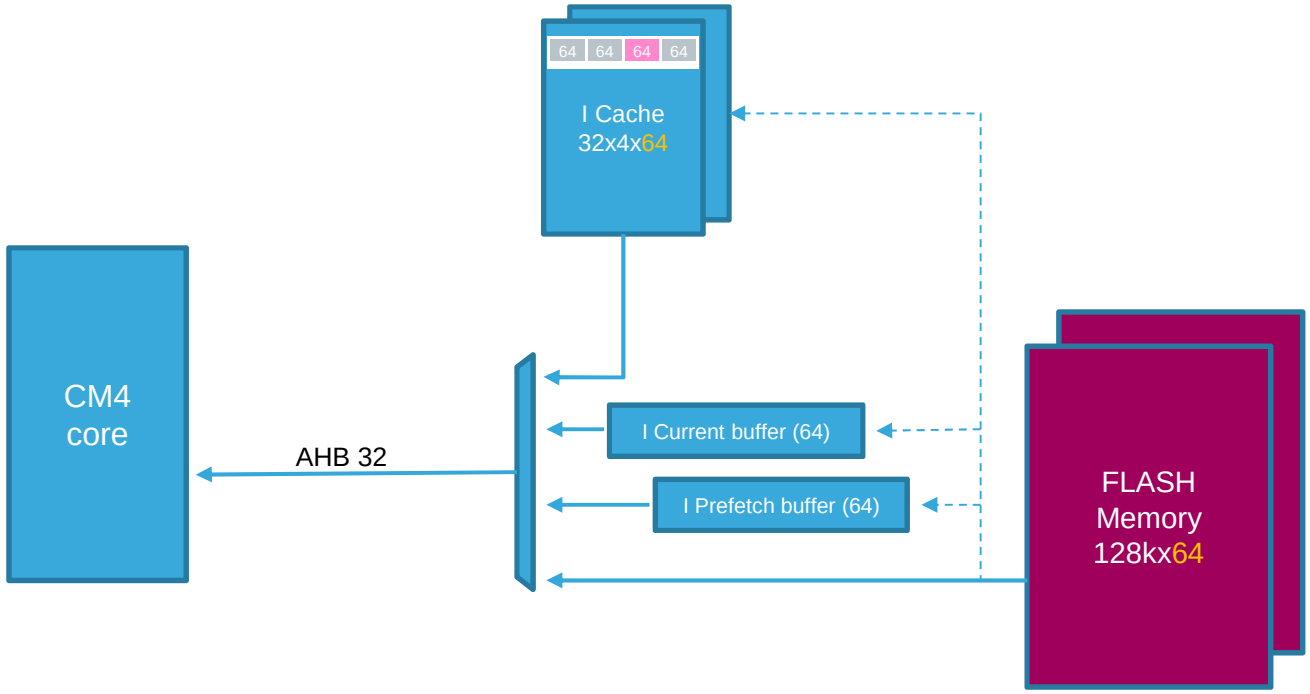
Mode	STM32L15x	STM32L4x
CPU	CortexM3	CortexM4 + FPU
Flash I/F	Prefetch	ART
Frequency	32MHz	80MHz
Performance	~35 DMIPS	~**100 DMIPS** (no loss)



life.augmented

ART overview

- Instruction cache = 32 lines of 4x64 bits (1KB)
- Data cache = 8 lines of 4x64 bits (256 B)
- Best tradeoff between cache size, Power and Dhrystone/CM performances



life.augmented

Flash protections

Flexible Protections configurable with option bytes :

- Readout protection (RDP)
  - Forbids access to Flash/SRAM2/Backup registers by:
    - Debug interface (JTAG/SWD)
    - Boot from SRAM1
    - Bootloader
- Proprietary Code Protection (PCROP) with 64-bit granularity
  - Used to protect specific code area from any read or write access
  - The code can only be executed.
- Write Protection (WRP) with 2-KByte granularity
  - Used to protect specific code area from unwanted write/erase

Readout Protections

- Readout protection Level 0 : No read protection
  - All operations are possible in all boot configurations.
- Readout protection Level 1
  - **User mode:** Code executing in user mode can access main Flash memory, option bytes, RTC backup registers and SRAM2 with all operations.
  - **Debug, boot RAM and boot loader modes:** The main Flash memory, backup registers and SRAM2 are totally inaccessible in these modes, a simple read access generates a bus error and a Hard Fault interrupt.
  - Un-protection, Level 1 to Level 0:
    - Flash memory is mass-erased, RTC backup registers and SRAM2 are cleared
    - If Option bit ***PCROP_RDP*** is set, the PCROP-protected area is not erased
- Readout protection Level 2 (No debug)
  - All protections provided by **Level 1** are **active**.
  - **RAM boot, System memory boot and all debug features are disabled**
  - Option bytes can no longer be changed in user mode.
  - Un-protection is not possible. It is an irreversible operation

life.augmented

Proprietary Code (PCROP) / Write Protections

- When enabled, PCROP area is protected against all D-code bus accesses
  - The PCROP regions are **execute-only**
  - 1 area per bank, 64-bit granularity.
    - PCROP area can be increased but never decreased
    - Only way to deactivate PCROP is to change RDP from Level 1 to Level 0

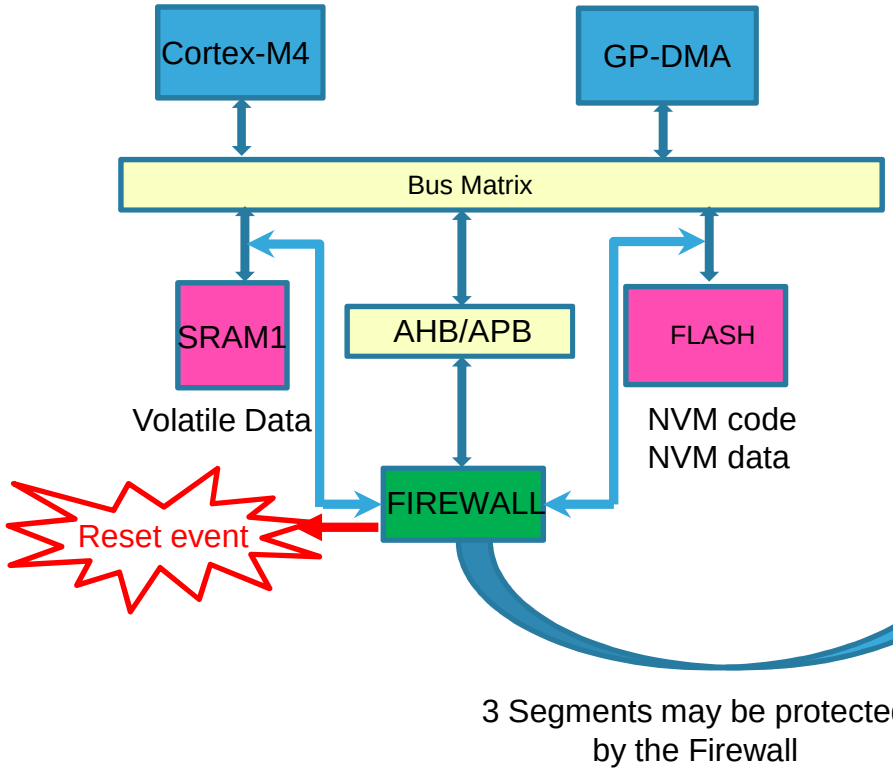
- Option bit **PCROP_RDP**
  - When DISABLED: PCROP content is erased when RDP is changed from Level 1 to Level 0. PCROP_RDP is locked in this state.
  - When ENABLED: PCROP content is preserved when RDP is changed from Level 1 to Level 0.

- Flash Write Protection:
  - Write-protected area is protected against erasing and programming.
  - 2 areas per bank, 2-KByte granularity.

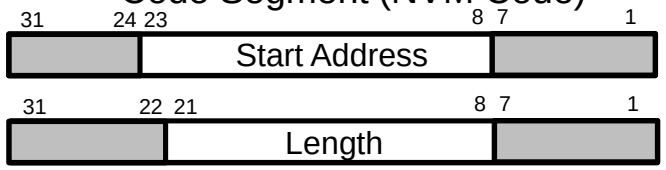
life.augmented

STM32 Firewall (FW)

- The FIREWALL is made to protect parts of code/data (volatile and non volatile) from access from the rest of the code executed outside of the protected area.

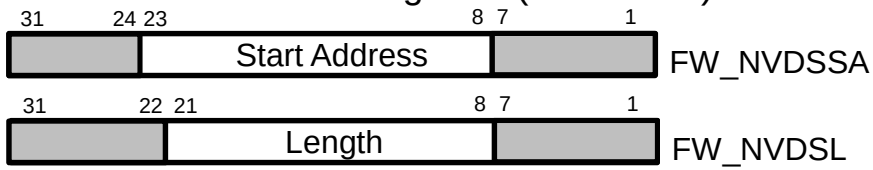


Code Segment (NVM Code)



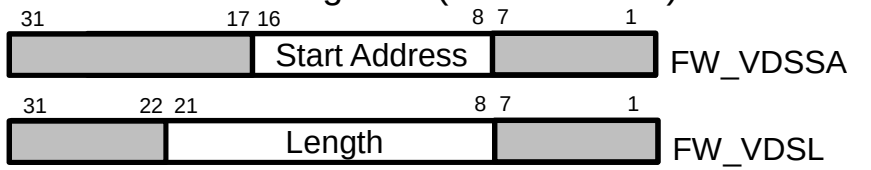
Protection : Code fetch

Non-Volatile Data Segment (NVM Data)

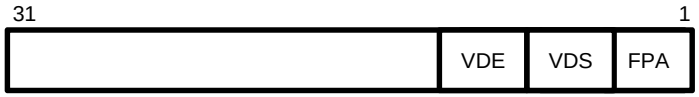


Protection : Data

Volatile Data Segment (SRAM1 Data)



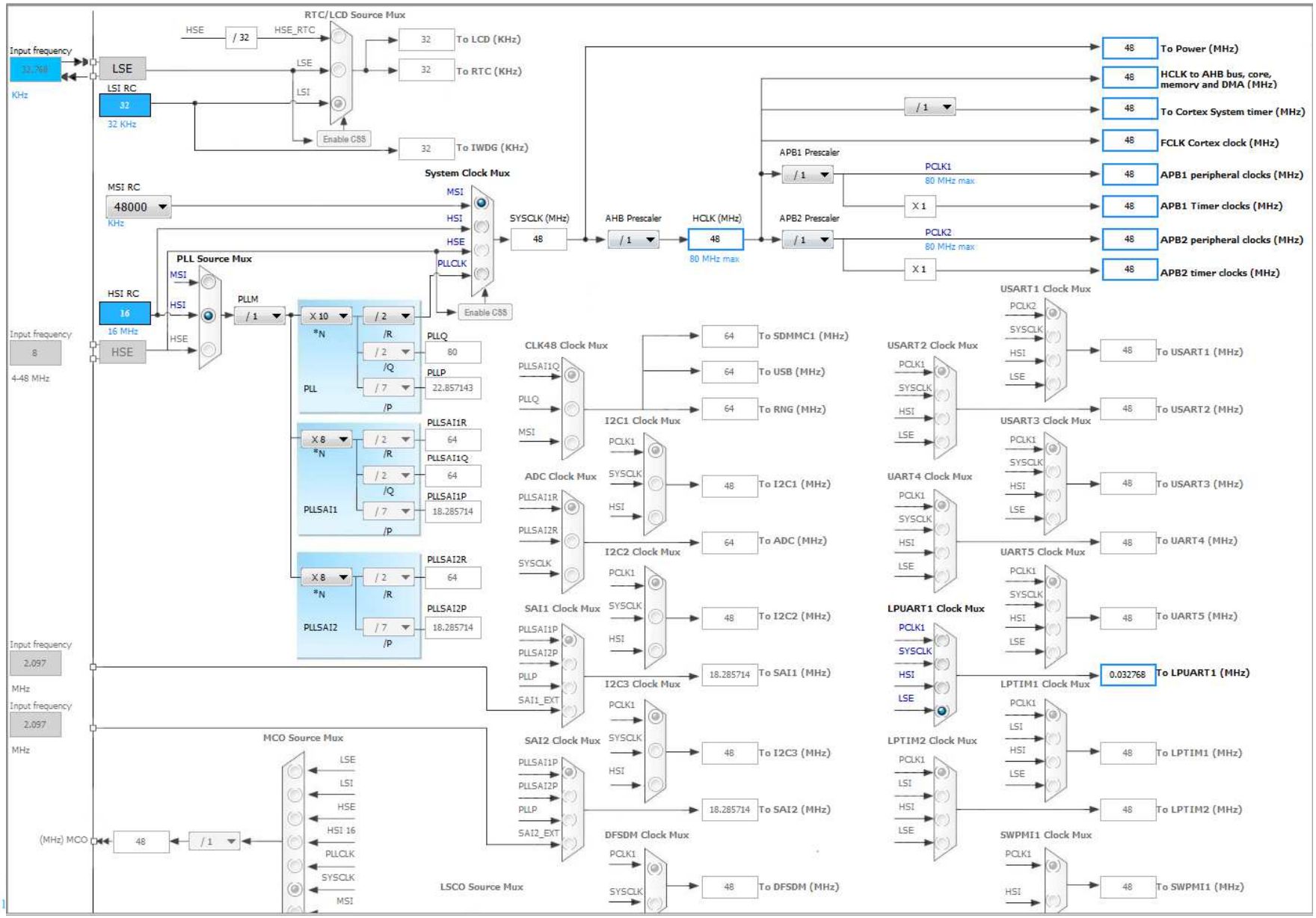
Protection: - Code fetch if SRAM1 is executable (not shared)
- Data



FPA : Pre-alarm bit to control the exit point of the protected code
VDS : SRAM1 protected segment is sharable with non protected code
VDE : SRAM1 is executable into the protected volatile data segment

life.augmented

Clock Tree


RTC/LCD Source Mux
HSE
/ 32
HSE_RTC
LSE
LSI
Enable CSS
32
To LCD (KHz)
To RTC (KHz)
To IWDG (KHz)
Input frequency
LSE
LSI RC
32 KHz
System Clock Mux
MSI RC
48000
MSI
HSI
HSE
PLLCLK
SYSCLK (MHz)
AHB Prescaler
HCLK (MHz)
80 MHz max
APB1 Prescaler
APB2 Prescaler
PCLK1
PCLK2
X 1
To Power (MHz)
HCLK to AHB bus, core, memory and DMA (MHz)
To Cortex System timer (MHz)
FCLK Cortex clock (MHz)
APB1 peripheral clocks (MHz)
APB1 Timer clocks (MHz)
APB2 peripheral clocks (MHz)
APB2 timer clocks (MHz)
PLL Source Mux
HSI RC
16 MHz
PLLM
PLL
PLLSAI1
PLLSAI2
*N
/R
/Q
/P
PLLQ
PLLP
PLLSAI1R
PLLSAI1Q
PLLSAI1P
PLLSAI2R
PLLSAI2P
Input frequency
4-48 MHz
CLK48 Clock Mux
To SDMMC1 (MHz)
To USB (MHz)
To RNG (MHz)
I2C1 Clock Mux
ADC Clock Mux
I2C2 Clock Mux
SAI1 Clock Mux
I2C3 Clock Mux
SAI2 Clock Mux
DFSDM Clock Mux
To I2C1 (MHz)
To ADC (MHz)
To I2C2 (MHz)
To SAI1 (MHz)
To I2C3 (MHz)
To SAI2 (MHz)
To DFSDM (MHz)
USART1 Clock Mux
USART2 Clock Mux
USART3 Clock Mux
UART4 Clock Mux
UART5 Clock Mux
LPUART1 Clock Mux
LPTIM1 Clock Mux
LPTIM2 Clock Mux
SWPMI1 Clock Mux
To USART1 (MHz)
To USART2 (MHz)
To USART3 (MHz)
To UART4 (MHz)
To UART5 (MHz)
To LPUART1 (MHz)
To LPTIM1 (MHz)
To LPTIM2 (MHz)
To SWPMI1 (MHz)
MCO Source Mux
(MHz) MCO
LSCO Source Mux

Clocks: MSI (Multi-Speed Internal)

- MSI = clock at startup from Reset, Standby or Shutdown modes.
- 12 Programmable frequency ranges:

  100 kHz, 200 kHz, 400 kHz, 800 kHz, 1 MHz, 2 MHz, **4 MHz (reset value)**, 8 MHz, 16 MHz, 24 MHz, 32 MHz, 48 MHz.
- After Standby: Frequency selected from 1, 2, 4 or 8 MHz with **MSISRANGE** in RCC_CSR register.
- Normal mode and **PLL-mode (=auto-calibration with LSE)**

  PLL-mode : **allows USB FS device functionality** (0.25% accuracy)
- Factory and user trimmed

Clocks: HSI(High-Speed Internal)

- **HSI** 16MHz, factory and user trimmed
  - Selectable as wakeup clock from STOP1 / STOP2
- Can be automatically woken up when exiting Stop modes
  - Can be used as I2C/USART/LPUART source upon wakeup

HSI vs. MSI (design spec)

	MSI		HSI
	Normal mode	PLL-mode	
Accuracy (max)	Over temperature: ±6%	Better than 0.25%	0.3% after trim
	Over voltage: From 100 to 800 kHz : ±1% From 1 to 8 MHz : ±2% From 16 to 48 MHz : ±4%		
Consumption (typ)	100 kHz : 0.5 µA		130 µA
	800 kHz : 2µA		
	1 MHz : 5µA		
	8 MHz : 20 µA		
	16 MHz : 60 µA		
	48 MHz : 160 µA		
Startup time (typ)	100 kHz : 10 µs 48 MHz : 2.5 µs	1.25 ms for 5% of final freq.	0.9 µs

life.augmented

Clocks: HSE(High-Speed External)

- **HSE** 4-48MHz
  - External source (Bypass mode) up to 40 MHz
  - External Crystal/Ceramic resonator (4-48MHz)

- Clock Security System (CSS)
  - Automatic detection of HSE failure with
    - NMI generation
    - Break input to TIM1/TIM8/TIM15/TIM16/TIM17
  - Backup clock can be HSI or MSI

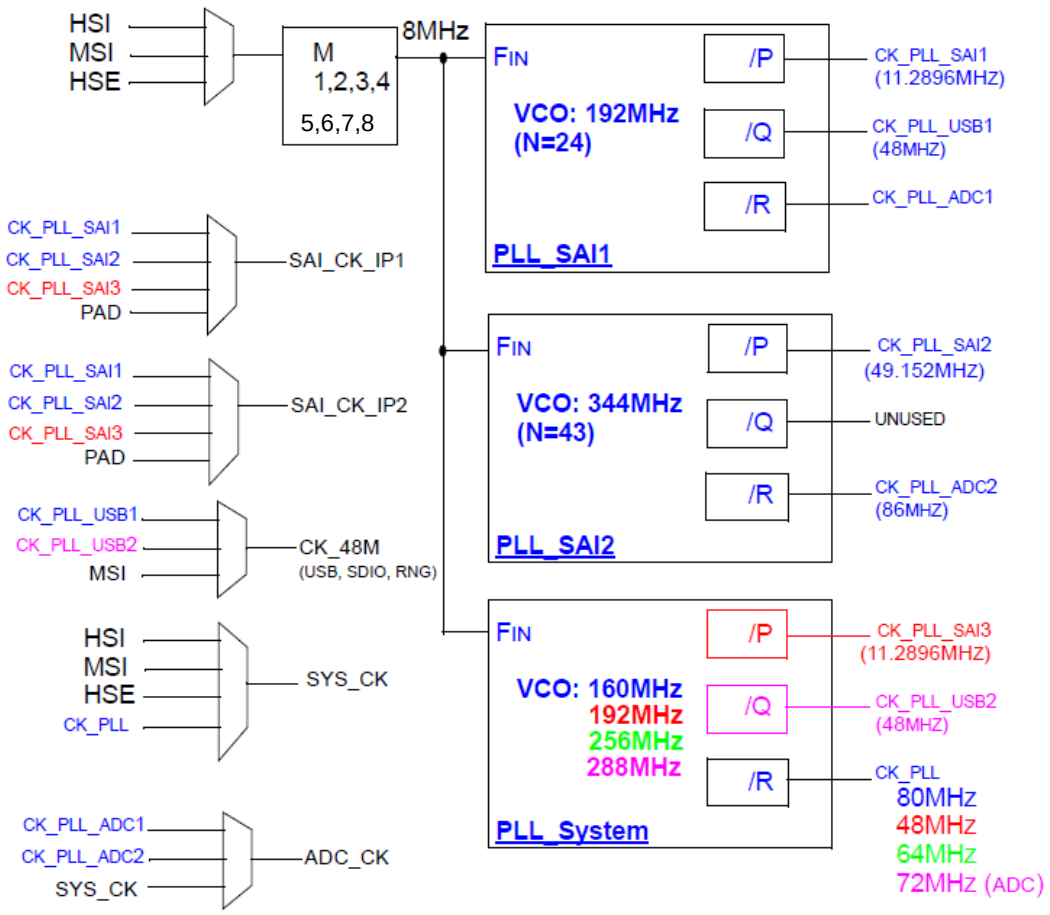
life.augmented

Clocks: PLL

- **3 PLL's**
  - Each with 3 independent outputs
  - PLL input freq = 4-16 MHz
  - PLL input can be MSI/HSI or HSE

f(VCO clock) =
(f(PLL clock input)/**PLLM**) × **PLLN**
f(PLL_P) = f(VCO clock) / **PLLP**
f(PLL_Q) = f(VCO clock) / **PLLQ**
f(PLL_R) = f(VCO clock) / **PLLR**

PLLM from 1 to 8
PLLN from 8 to 86
PLLP = 7 or 17
PLLQ = 2, 4, 6, 8
PLLR = 2, 4, 6, 8



PLLP from 2 to 31 in derivatives

life.augmented

Clocks: LSE (Low-Speed External)

- **LSE**: programmable amplifier driving capability (4 modes)

Mode	Crystal (max)	Consumption (typ)
Ultra low power	50kOhm/6pF	200 nA
Medium low driving	80kOhm/6pF	260 nA
Medium high driving	50kOhm/12.5pF	410 nA
High driving	80kOhm/12.5pF	540 nA

- Available in all low-power modes + VBAT

Clocks: LSI (Low-Speed Internal)

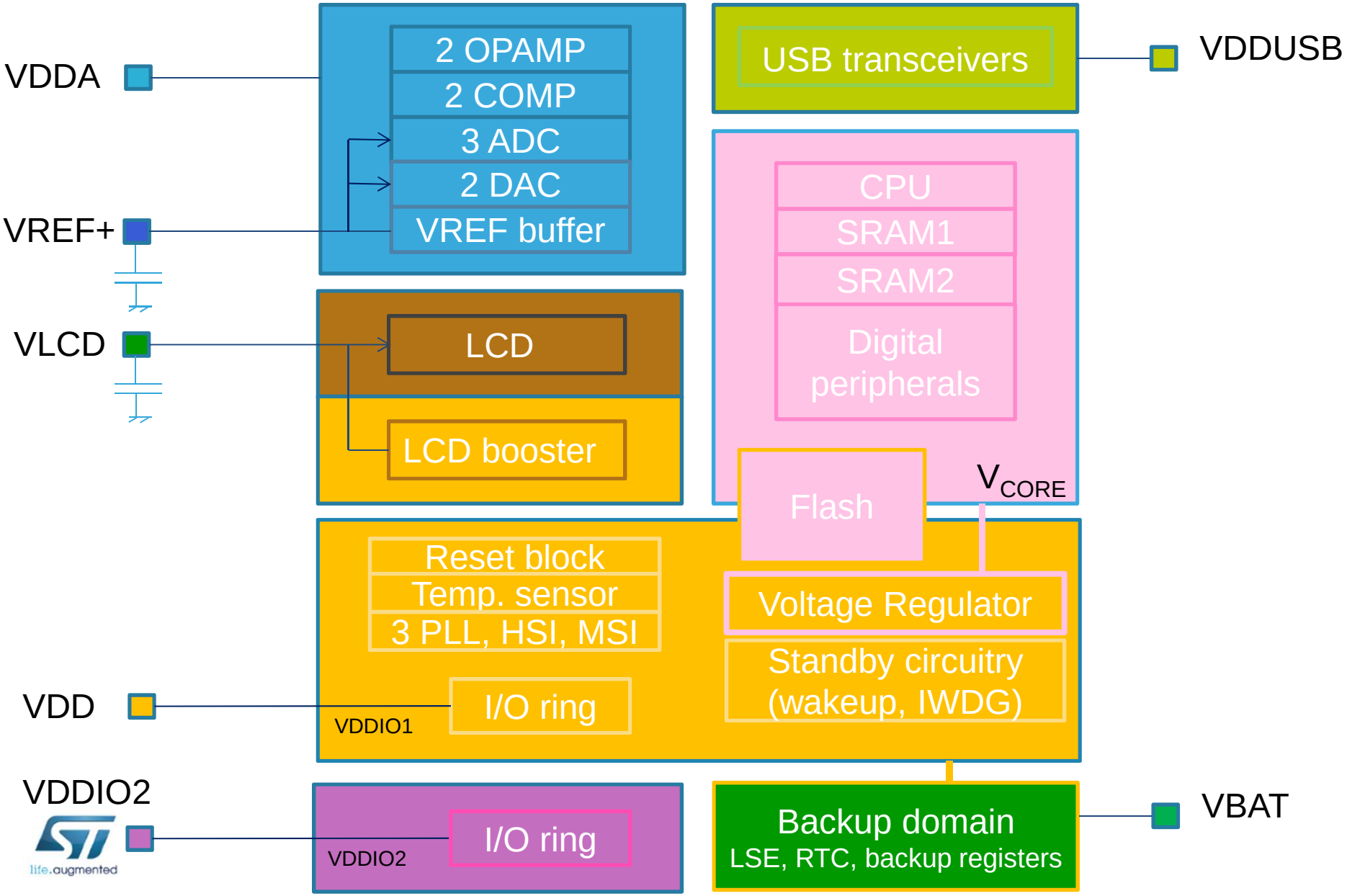
- **LSI** 32kHz

Parameters	STM32L15x	STM32L4x
Accuracy over parts	26..56KHz	**~8%**
Accuracy over temperature	-10% / +4% (0..85°C)	**2% (-40 125°C)**
Consumption (typ)	400nA	**110nA**

- Available in all low-power modes except Shutdown and VBAT

life.augmented

Power schemes (1/3)



Peripheral Voltage Monitor

- By default independent powers are electrically isolated and the features powered by them are not available
  - The power isolation must be removed by SW

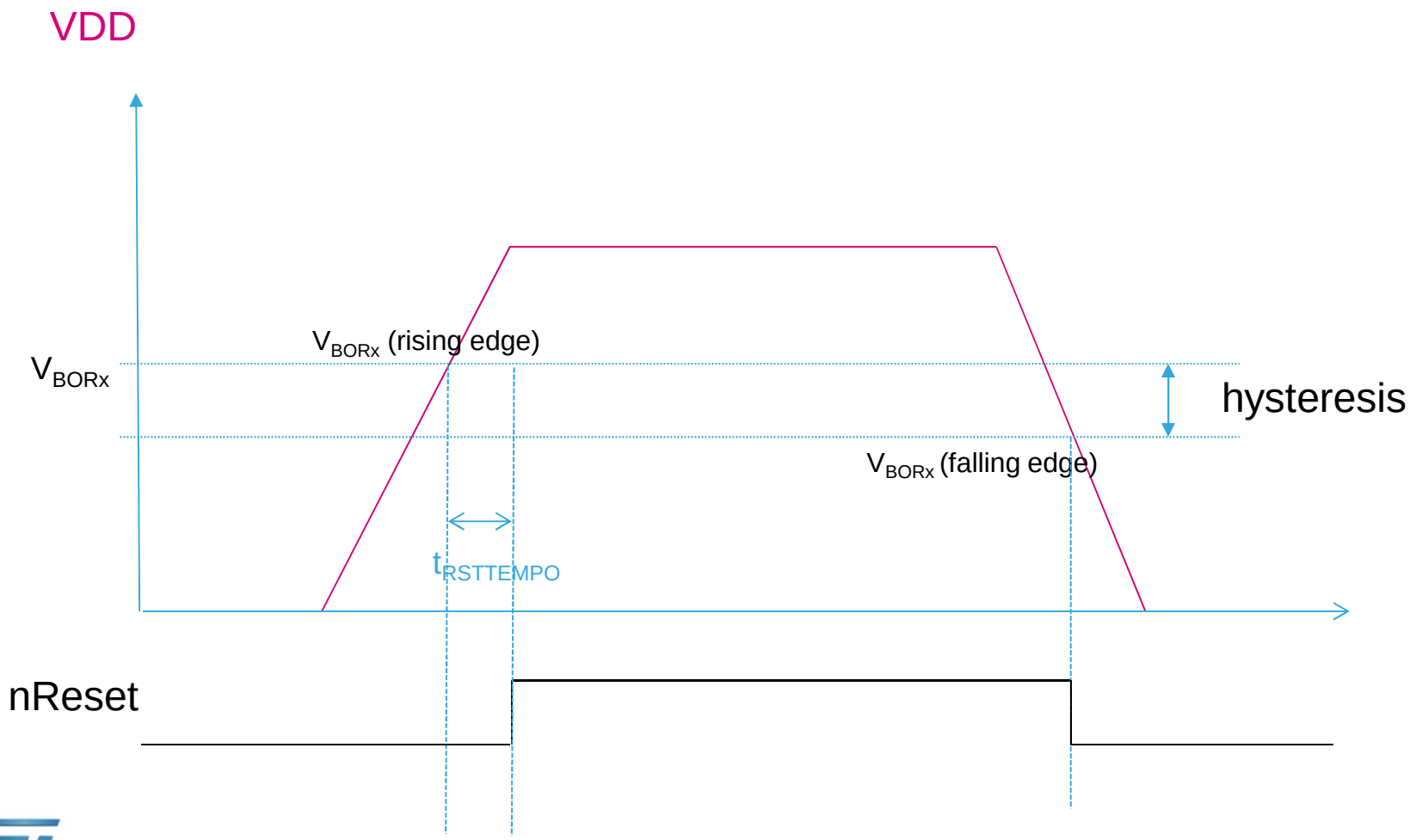
- Peripheral Voltage Monitor for VDDA, VDDUSB, VDDIO2:

PVM	Power supply	PVM threshold	EXTI line
PVM1	V_{DDUSB}	VPVM1 = 1.2 V	35
PVM2	V_{DDIO2}	VPVM2 = 0,9 V	36
PVM3	V_{DDA}	VPVM3 = 1.65 V	37
PVM4	V_{DDA}	VPVM4 = 1.82 V	38

life.augmented

Power supply Supervisor

BOR complies with all VDD rise/fall time = no constraint on power supply shape

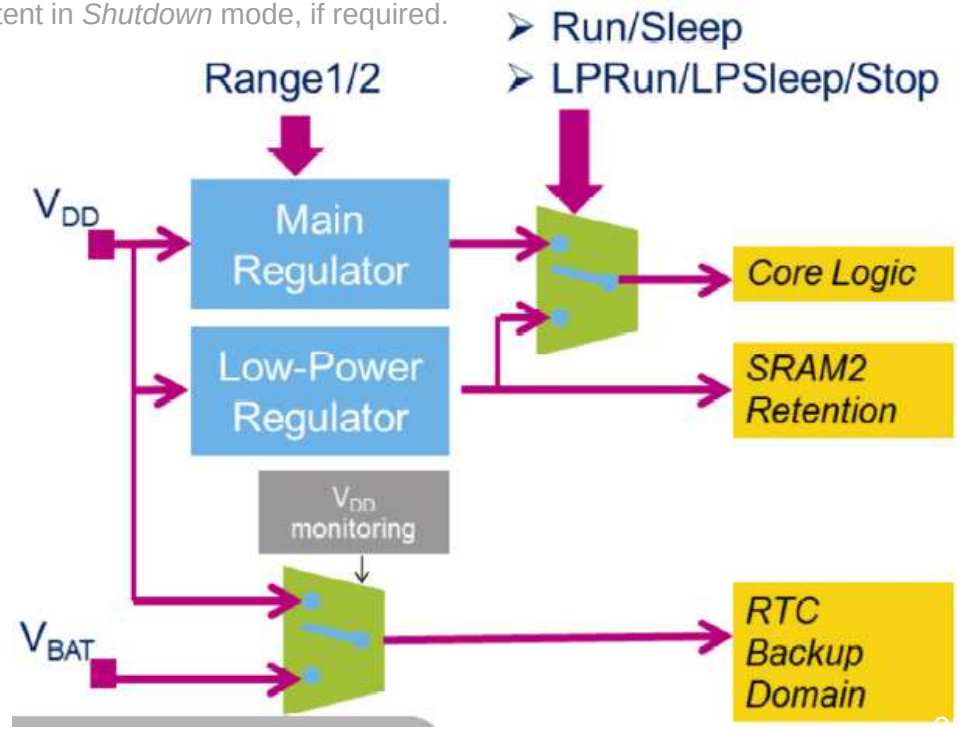


life.augmented

V_{CORE} Voltage Regulator

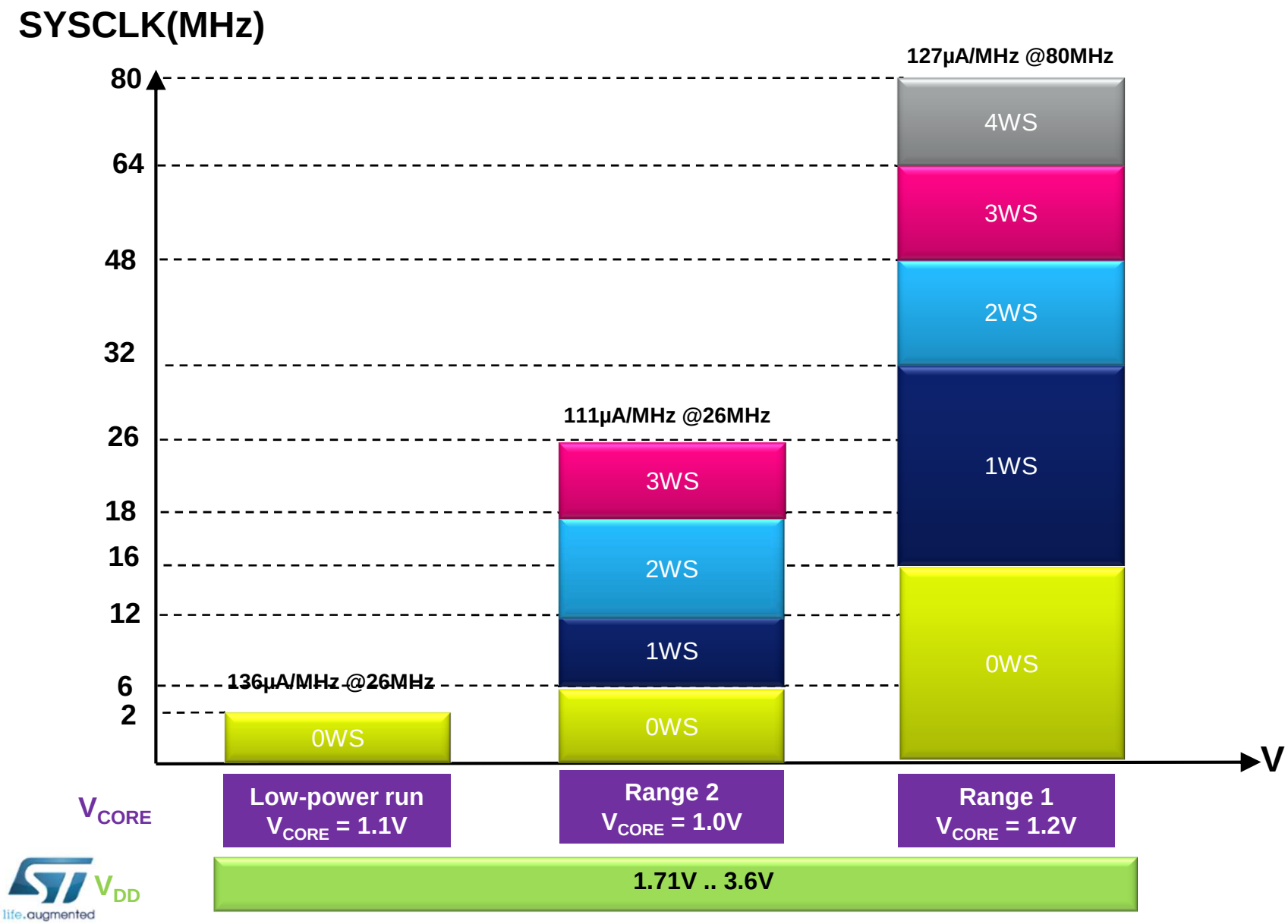
- Dynamic Voltage Scaling optimizes performance vs power
  - V_{CORE} Voltage Range 1 = 1.2V (Up to 80MHz)
  - V_{CORE} Voltage Range 2 = 1.0V (Up to 26MHz)

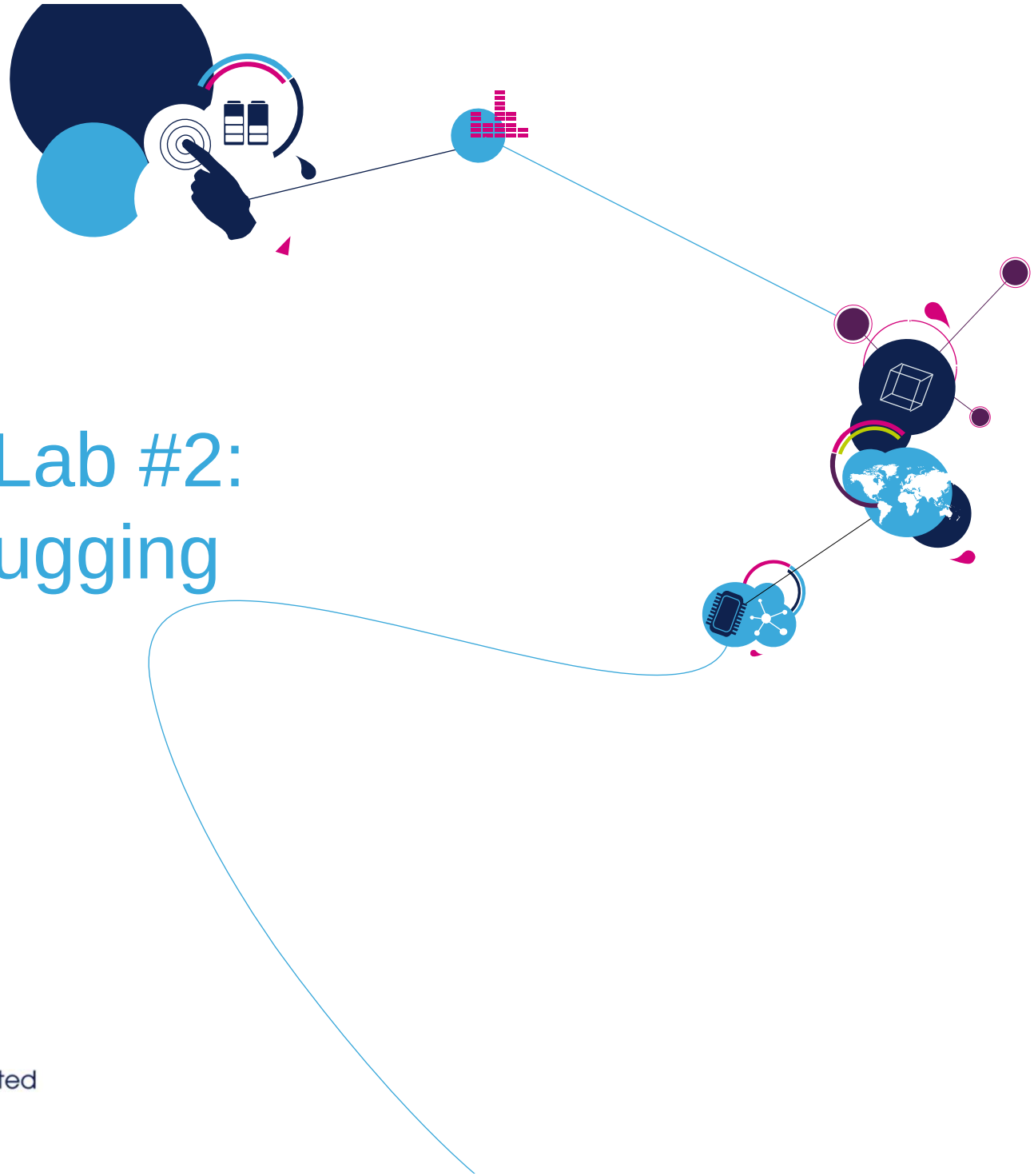
- V_{CORE} powered by main regulator (MR) or low-power regulator (LPR)
  - MVR for Run and Sleep modes.
  - LPR for LP run, LP sleep and STOP1/STOP2 modes.
  - Regulators OFF in Standby and Shutdown mode.
    - However LPR remains ON to preserve SRAM2 content in *Shutdown* mode, if required.



life.augmented

Dynamic voltage scaling in Run mode

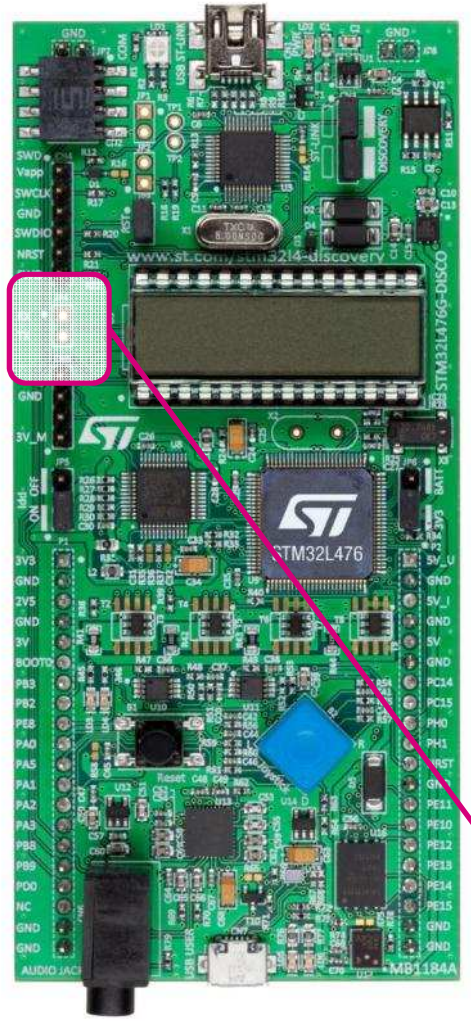




Hands-On Lab #2: printf() debugging

life.augmented

Hands-On Lab #2 – printf() debugging



Adding to the existing CubeMX project, we will add USART2 debug via the ST-LINK Virtual-COM port

Set up additional GPIO / Clocks:
PD5 – USART2, “VirtualCOM-TX” – Alt. Fn. Push/Pull
PD6 – USART2, “VirtualCOM-RX” – Alt. Fn. Push/Pull
PA0 – GPIO_EXTI0: External Interrupt on Rising Edge Input
USART2 Clock = PCLK1 (80MHz)

USART2 settings:
Asynchronous Mode - 9600 N/8/1, No HW Flow control
Tx/Rx, 16-sample oversampling
No advanced features

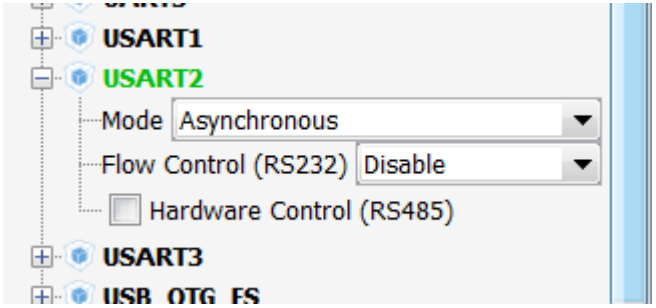
User Code HAL function calls required:
See **lab2_printf_debug.c** in **Lab2_Printf** directory

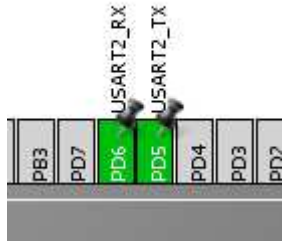
Also Need Terminal emulator (Hyperterm, etc)

USART2 is routed to the ST-LINK’s USART, and brought via the USB Virtual-COM port class (SB13/16 have been soldered)

GPIO Configuration additions

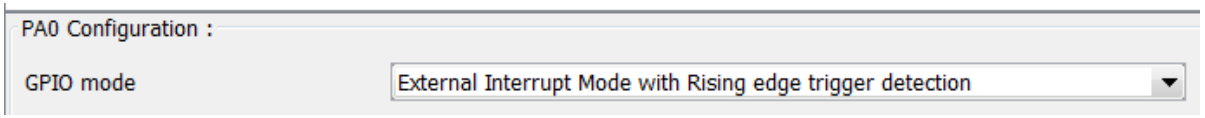
- Expand the USART2 dialog, and select **Asynchronous** mode:





- Use PD5 & PD6 for Tx / Rx pins:
  - These are the alternate mapping pins (PA2/3 are default)

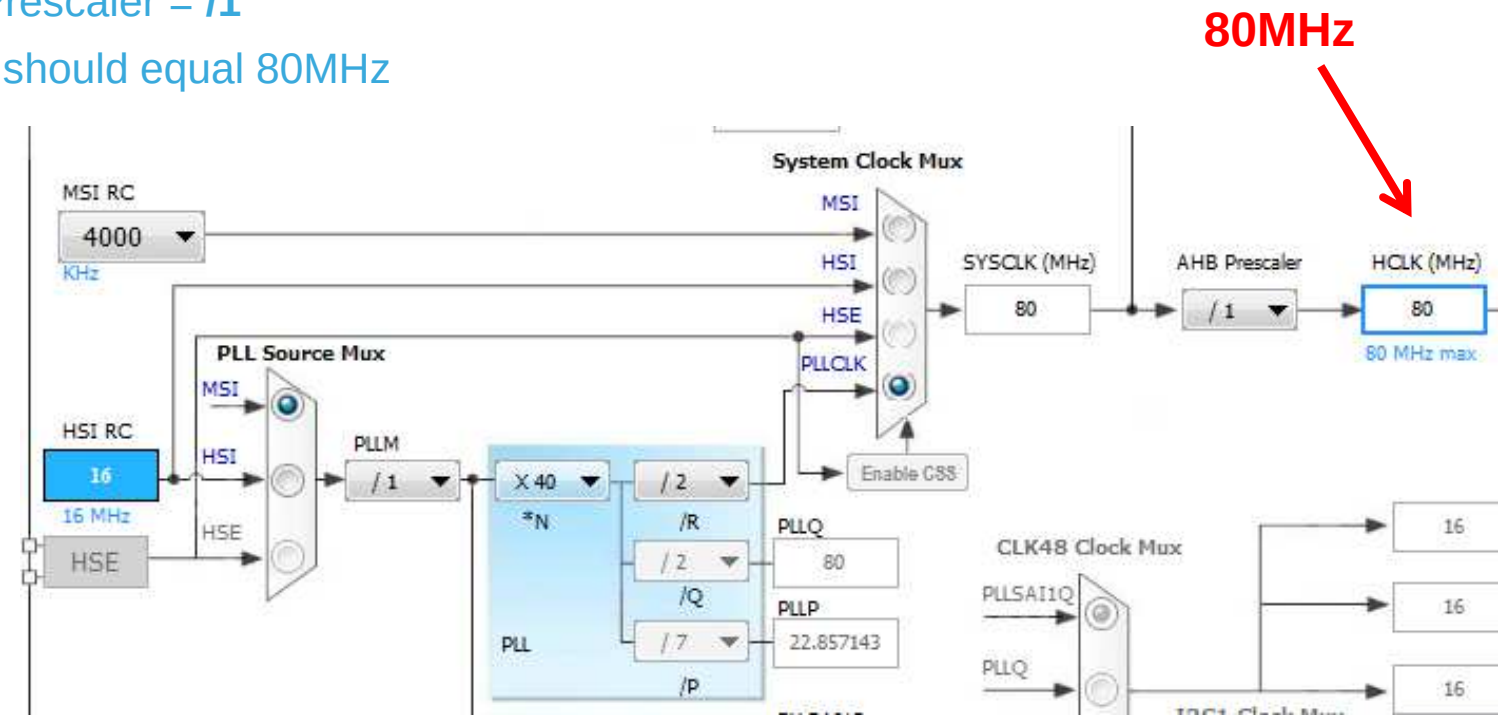
- Use PA0 in External Interrupt Mode, Rising edge trigger:



life.augmented

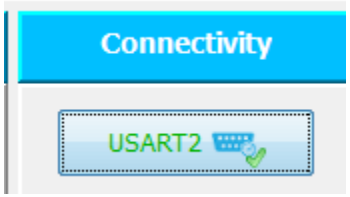
Clock Configuration

- We will run the STM32L4 at 80MHz for this lab
- Click on the ***Clock Configuration*** tab
  - Set the **PLL Source Mux** to **MSI (MSI RC = 4000 KHz)**
  - Set the **System Clock Mux** to **PLLCLK**
  - Use PLLM=**/1**, *N = **x40**, /R=**/2**
  - AHB Prescaler = **/1**
  - HCLK should equal 80MHz

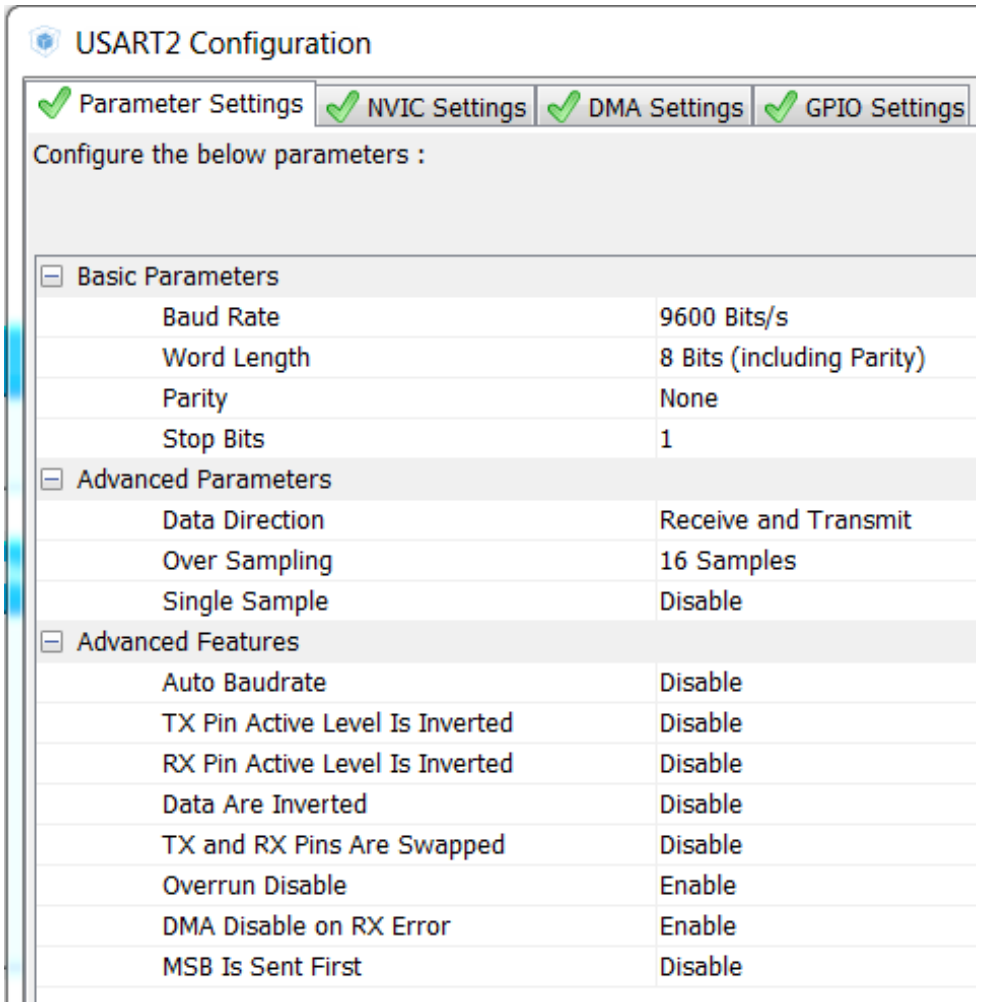


USART2 Configuration

- Click on the Configuration tab, and select USART2:



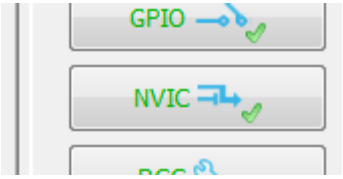
- Parameter Settings tab:
  - 9600Bits/s
  - 8-bit word length
  - No parity bit
  - 1 Stop bit
  - Rx & Tx data
  - 16-clock oversampling
  - No advanced feature settings needed
  - No NVIC or DMA settings used



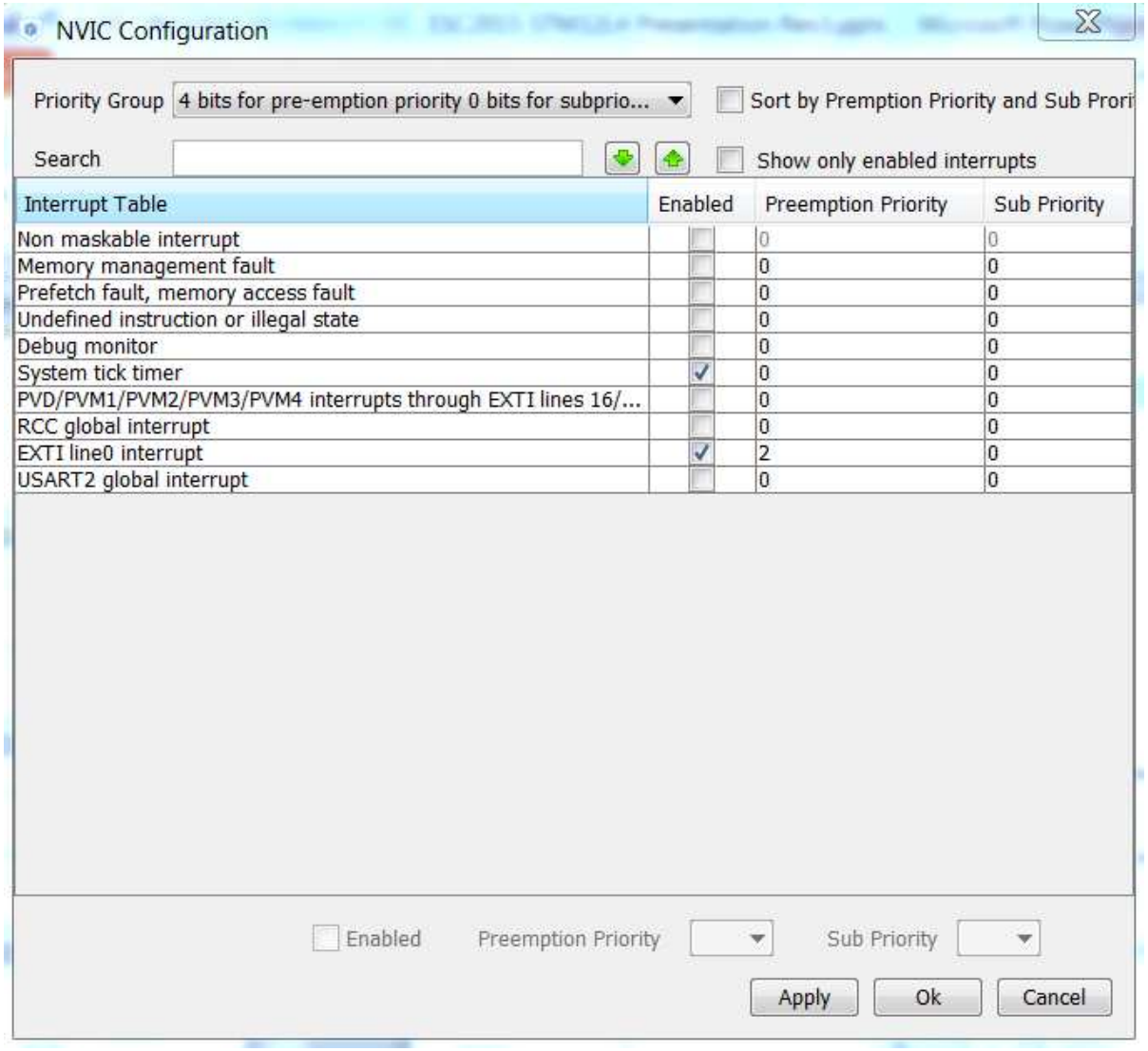
life.augmented

NVIC Configuration

- Click on the Configuration tab, and select NVIC:



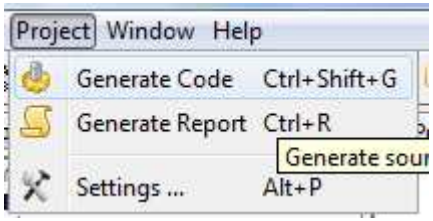
- Enable EXTI line0
- Set Preemption Priority to 2
- Click ‘OK’

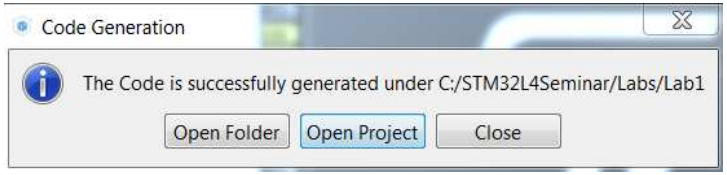


life.augmented

Regenerate Source Code for Lab1

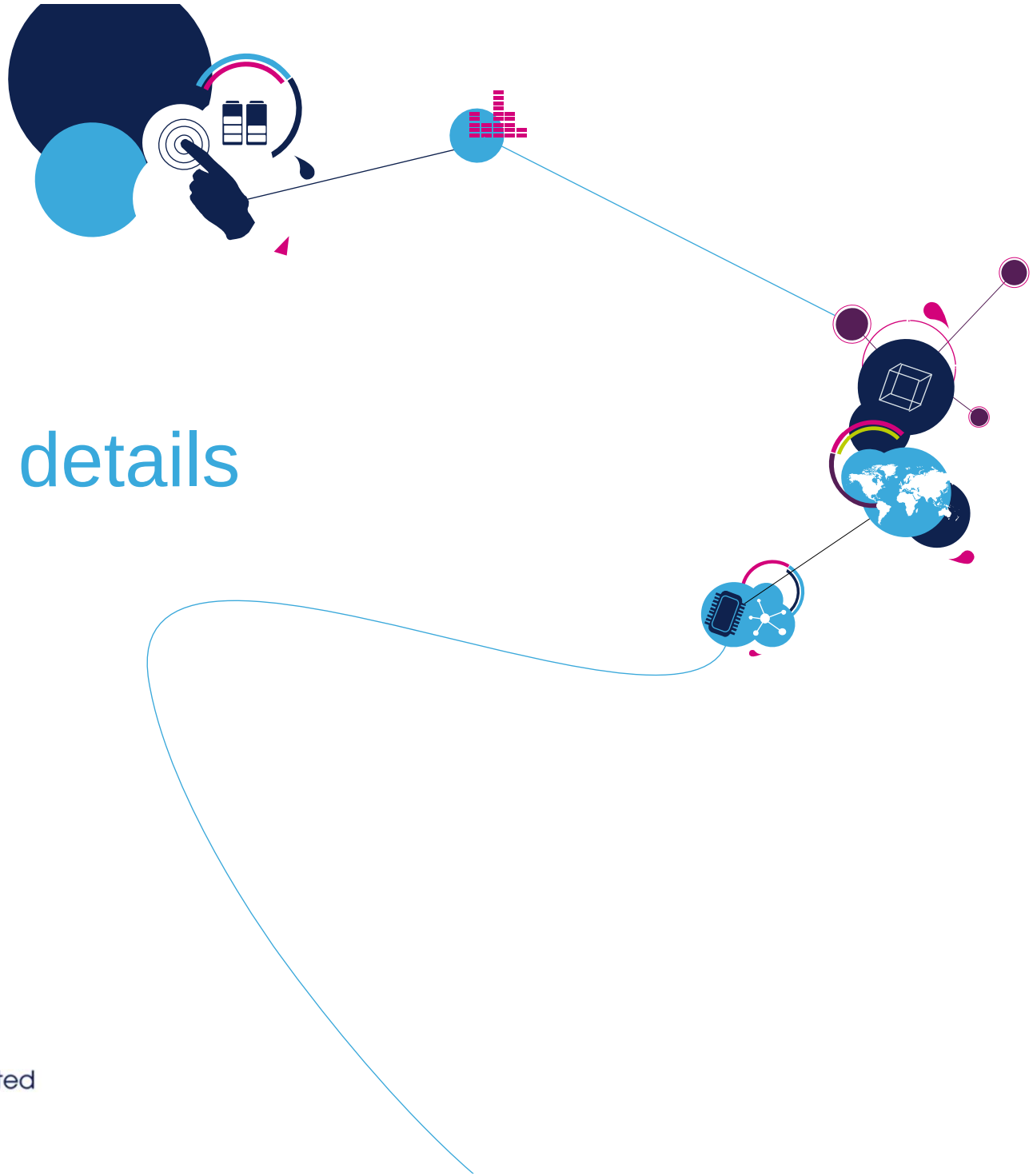
- **Generate Code** (Ctrl + Shift + G)
- **Open Project**





- Copy/Paste needed code bits for Hands-On Lab #2 into main.c:
  - C:\STM32L4Seminar\Labs**lab2_printf_debug.c**
    - **Add code bits between:** **/* USER CODE BEGIN PV */** **/* USER CODE END PV */**
    - **Add code bits between:** **/* USER CODE BEGIN 2 */** **/* USER CODE END 2 */**
    - **Add code bits between:** **/* USER CODE BEGIN WHILE */** **/* USER CODE END WHILE */**
    - **Add code bits between:** **/* USER CODE BEGIN 4 */** **/* USER CODE END 4 */**
  - Open a terminal emulator, using USART2 settings, Virtual COM port xx
  - Rebuild, Program/Debug, Run!

life.augmented



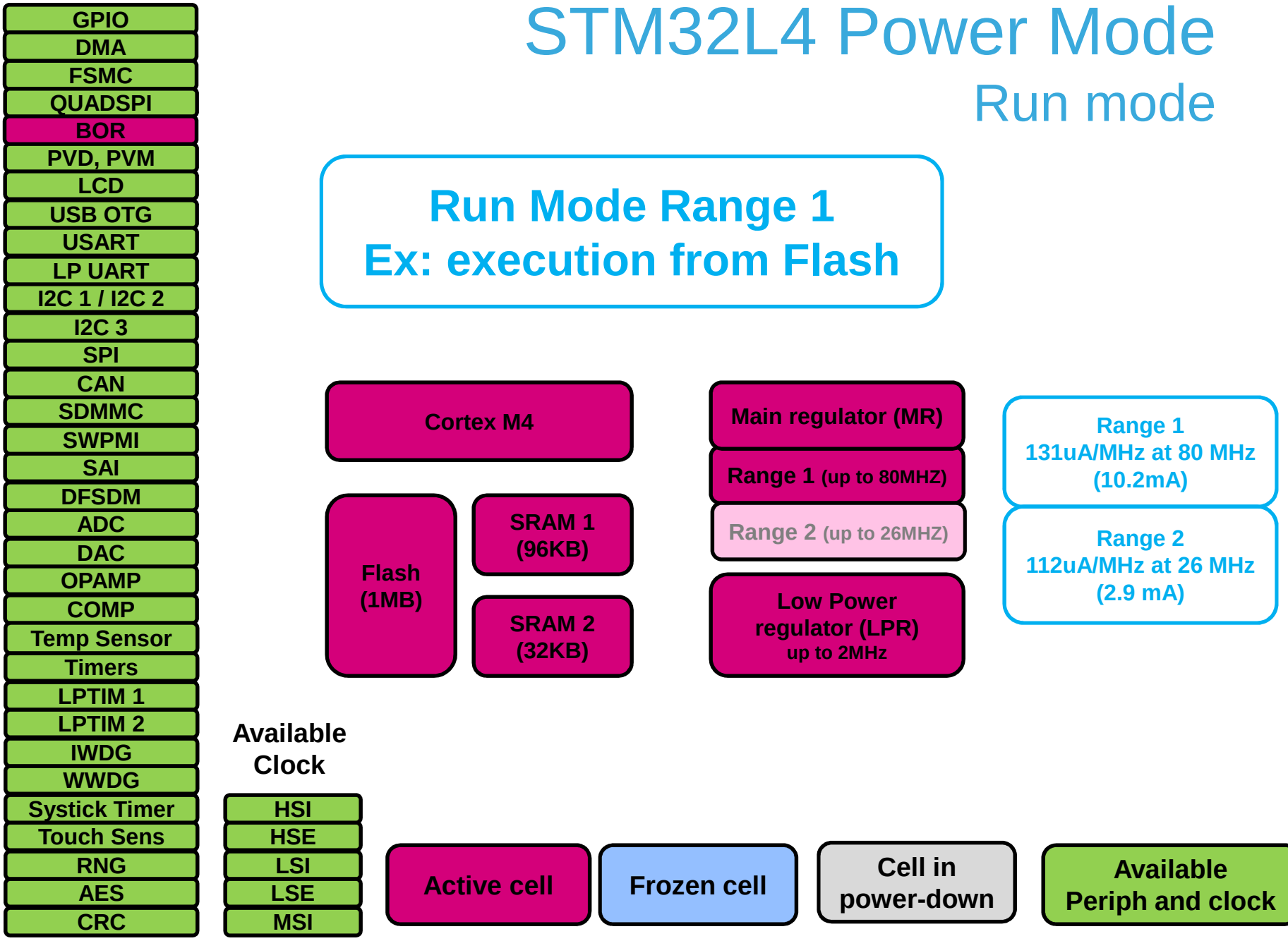
STM32L4
Low Power details

life.augmented

STM32L4 Power Mode

Run mode

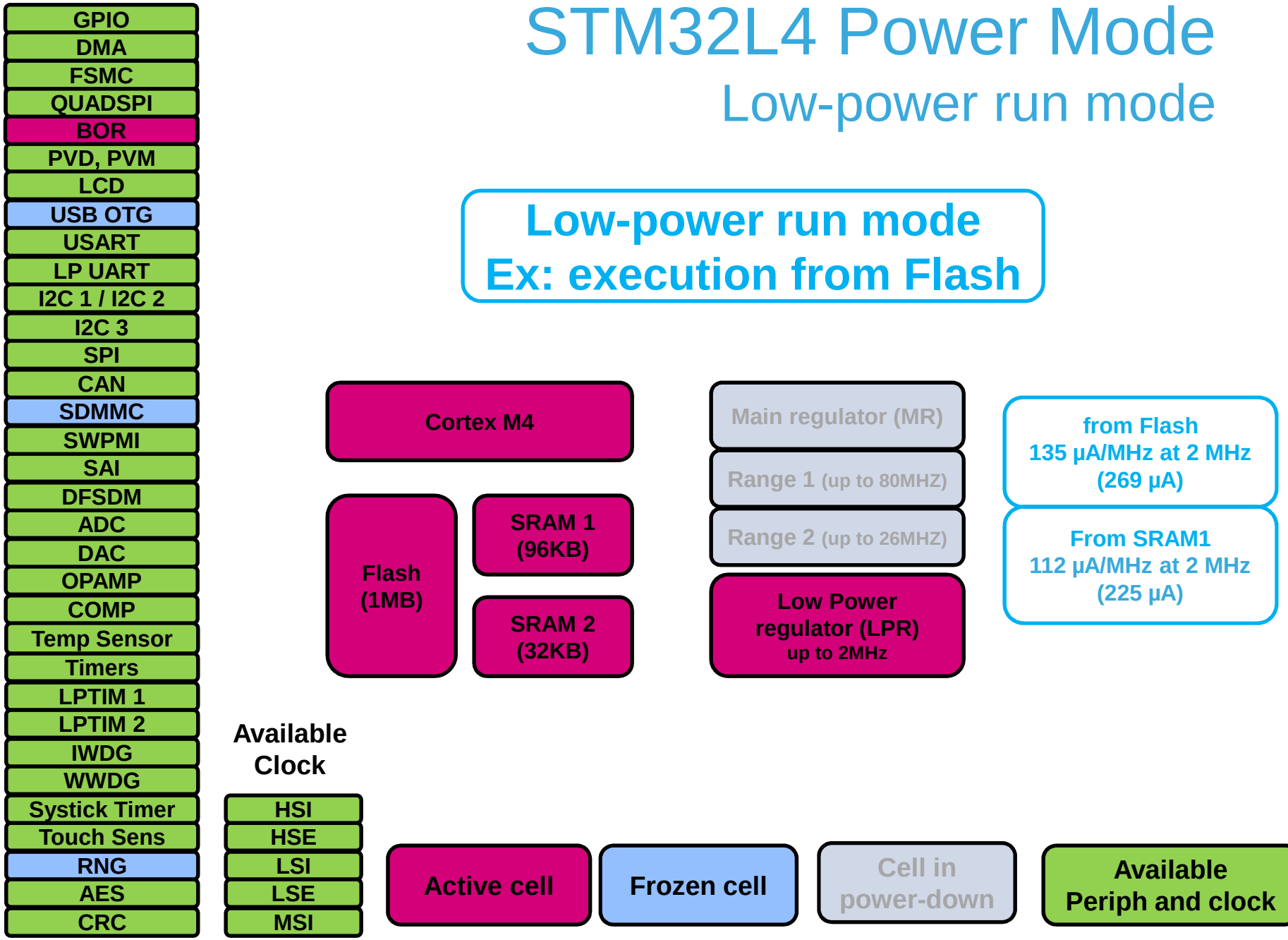
Run Mode Range 1
Ex: execution from Flash



STM32L4 Power Mode

Low-power run mode

Low-power run mode
Ex: execution from Flash



STM32L4 Power Mode

Sleep mode

Sleep Mode Range 1
Ex: Flash ON, SRAMs ON (default)

Cortex M4

Flash (1MB)

SRAM 1 (96KB)

SRAM 2 (32KB)

Main regulator (MR)

Range 1 (up to 80MHZ)

Range 2 (up to 26MHZ)

Low Power regulator (LPR) up to 2MHz

Range 1
37 µA/MHz at 80 MHz
(2.96 mA)

Range 2
35 µA/MHz at 26 MHz
(0.92 mA)

Available Peripheral

- GPIO
- DMA
- FSMC
- QUADSPI
- BOR
- PVD, PVM
- LCD
- USB OTG
- USART
- LP UART
- I2C 1 / I2C 2
- I2C 3
- SPI
- CAN
- SDMMC
- SWPMI
- SAI
- DFSDM
- ADC
- DAC
- OPAMP
- COMP
- Temp Sensor
- Timers
- LPTIM 1
- LPTIM 2
- IWDG
- WWDG
- Systick Timer
- Touch Sens
- RNG
- AES
- CRC

Available Clock

- HSI
- HSE
- LSI
- LSE
- MSI

Active cell

Frozen cell

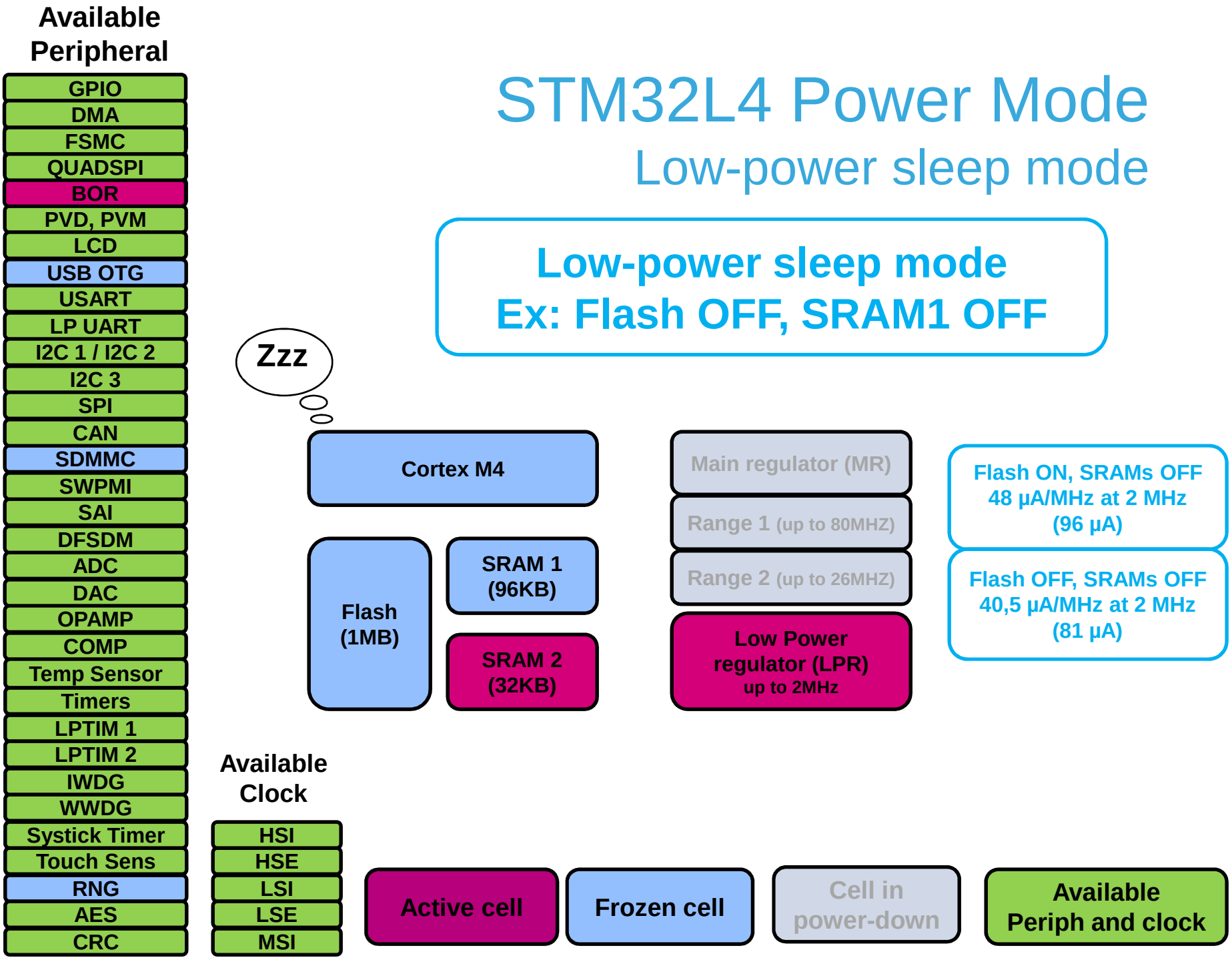
Cell in power-down

Available Periph and clock

STM32L4 Power Mode

Low-power sleep mode

Low-power sleep mode
Ex: Flash OFF, SRAM1 OFF



STM32L4 Power Mode

Stop 1 Mode

Stop 1 w/ RTC on LSE quartz → **7.1 µA @3.0V**
6.9 µA @1.8V

Available Peripheral

I/Os kept, and configurable

- GPIO
- DMA
- FSMC
- QSPI
- BOR
- PVD, PVM
- LCD
- USB OTG
- USART
- LP UART
- I2C 1 / I2C 2
- I2C 3
- SPI
- CAN
- SDMMC
- SWPMI
- SAI
- DFSDM
- ADC
- DAC
- OPAMP
- COMP
- Temp Sensor
- Timers
- LPTIM 1
- LPTIM 2
- IWDG
- WWDG
- Systick Timer
- Touch Sens
- RNG
- AES
- CRC

Zzz

- Cortex M4
- Flash (1MB)
- SRAM 1 (96KB)
- SRAM 2 (32KB)

Available Clock

- HSI
- HSE
- LSI
- LSE
- MSI

- Main regulator (MR)
- Low Power regulator (LPR)
- Backup domain
  - Backup Register (32x32-bits)
  - RTC

Wake-up event

- NRST
- BOR
- PVD
- PVM
- RTC + Tamper
- LCD
- USB OTG
- USART
- LP UART
- I2C 1 / I2C 2
- I2C 3
- SWPMI
- COMP
- LPTIM 1
- LPTIM 2
- IWDG
- GPIOs

6us wake-up from Flash
4us wake-up from RAM

STM32L4 Power Mode

Stop 2 Mode

Stop 2 w/ RTC on LSE quartz → **1.67 µA @3.0V**
1.43 µA @1.8V

Available Peripheral

I/Os kept, and configurable

- GPIO
- DMA
- FSMC
- QSPI
- BOR
- PVD, PVM
- LCD
- USB OTG
- USART
- LP UART
- I2C 1 / I2C 2
- I2C 3
- SPI
- CAN
- SDMMC
- SWPMI
- SAI
- DFSDM
- ADC
- DAC
- OPAMP
- COMP
- Temp Sensor
- Timers
- LPTIM 1
- LPTIM 2
- IWDG
- WWDG
- Systick Timer
- Touch Sens
- RNG
- AES
- CRC

Zzz

- Cortex M4
- Flash (1MB)
- SRAM 1 (96KB)
- SRAM 2 (32KB)

Available Clock

- HSI
- HSE
- LSI
- LSE
- MSI

- Main regulator (MR)
- Low Power regulator (LPR)
- Backup domain
  - Backup Register (32x32-bits)
  - RTC

Wake-up event

- NRST
- BOR
- PVD
- PVM
- RTC + Tamper
- LCD
- LP UART
- I2C 3
- COMP
- LPTIM 1
- IWDG
- GPIOs

7us wake-up from Flash
5us wake-up from RAM

STM32L4 Power Mode

Standby Mode

Available Peripheral

- GPIO
- DMA
- FSMC
- QSPI
- **BOR**
- PVD, PVM
- LCD
- USB OTG
- USART
- LP UART
- I2C 1 / I2C 2
- I2C 3
- SPI
- CAN
- SDMMC
- SWPMI
- SAI
- DFSDM
- ADC
- DAC
- OPAMP
- COMP
- Temp Sensor
- Timers
- LPTIM 1
- LPTIM 2
- **IWDG**
- WWDG
- Systick Timer
- Touch Sens
- RNG
- AES
- CRC

I/Os can be configured
w/ or w/o pull-up
w/ or w/o pull-down

Standby
w/ SRAM2
w/o RTC

405 nA @ 3.0V
363 nA @1.8V

Zzz

Cortex M4

Main regulator (MR)

Flash (1MB)

SRAM 1 (96KB)

SRAM 2 (32KB)

Low Power regulator (LPR)

Backup domain

Backup Register (32x32-bits)

RTC

Available Clock

- HSI
- HSE
- **LSI**
- **LSE**
- MSI

Wake-up event

- **NRST**
- **BOR**
- **RTC + Tamper**
- **IWDG**
- **5 WKUP pins**

14 us wake-up

STM32L4 Power Mode

Standby Mode

Available Peripheral

- GPIO
- DMA
- FSMC
- QSPI
- **BOR**
- PVD, PVM
- LCD
- USB OTG
- USART
- LP UART
- I2C 1 / I2C 2
- I2C 3
- SPI
- CAN
- SDMMC
- SWPMI
- SAI
- DFSDM
- ADC
- DAC
- OPAMP
- COMP
- Temp Sensor
- Timers
- LPTIM 1
- LPTIM 2
- **IWDG**
- WWDG
- Systick Timer
- Touch Sens
- RNG
- AES
- CRC

I/Os can be configured
w/ or w/o pull-up
w/ or w/o pull-down

Standby w/ RTC on LSE quartz

650 nA @ 3.0V
433 nA@ 1.8V

Zzz

- Cortex M4
- Main regulator (MR)
- Flash (1MB)
- SRAM 1 (96KB)
- SRAM 2 (32KB)
- Low Power regulator (LPR)
- **Backup domain**
  - Backup Register (32x32-bits)
  - RTC

Available Clock

- HSI
- HSE
- **LSI**
- **LSE**
- MSI

Wake-up event

- **NRST**
- **BOR**
- **RTC + Tamper**
- **IWDG**
- **5 WKUP pins**

14 us wake-up

STM32L4 Power Mode

Standby Mode

Available Peripheral

- GPIO
- DMA
- FSMC
- QSPI
- **BOR**
- PVD, PVM
- LCD
- USB OTG
- USART
- LP UART
- I2C 1 / I2C 2
- I2C 3
- SPI
- CAN
- SDMMC
- SWPMI
- SAI
- DFSDM
- ADC
- DAC
- OPAMP
- COMP
- Temp Sensor
- Timers
- LPTIM 1
- LPTIM 2
- **IWDG**
- WWDG
- Systick Timer
- Touch Sens
- RNG
- AES
- CRC

I/Os can be configured
w/ or w/o pull-up
w/ or w/o pull-down

Standby ➡ **150 nA @ 3.0V**
114 nA @ 1.8V

Zzz

- Cortex M4
- Main regulator (MR)
- Flash (1MB)
- SRAM 1 (96KB)
- SRAM 2 (32KB)
- Low Power regulator (LPR)
- **Backup domain**
  - **Backup Register (32x32-bits)**
  - **RTC**

Available Clock

- HSI
- HSE
- **LSI**
- **LSE**
- MSI

Wake-up event

- **NRST**
- **BOR**
- **RTC + Tamper**
- **IWDG**
- **5 WKUP pins**

14 us wake-up

STM32L4 Power Mode

Shutdown Mode

Available Peripheral

- GPIO
- DMA
- FSMC
- QSPI
- BOR
- PVD, PVM
- LCD
- USB OTG
- USART
- LP UART
- I2C 1 / I2C 2
- I2C 3
- SPI
- CAN
- SDMMC
- SWPMI
- SAI
- DFSDM
- ADC
- DAC
- OPAMP
- COMP
- Temp Sensor
- Timers
- LPTIM 1
- LPTIM 2
- IWDG
- WWDG
- Systick Timer
- Touch Sens
- RNG
- AES
- CRC

I/Os can be configured
w/ or w/o pull-up
w/ or w/o pull-down
But floating when exit from Shutdown

Shutdown w/ RTC on LSE quartz ⇒ **550 nA @ 3.0V**
329 nA @ 1.8V

- Cortex M4
- Flash (1MB)
- SRAM 1 (96KB)
- SRAM 2 (32KB)

- Main regulator (MR)
- Low Power regulator (LPR)
- **Backup domain**
  - Backup Register (32x32-bits)
  - RTC

Available Clock

- HSI
- HSE
- LSI
- **LSE**
- MSI

Wake-up event

- NRST
- RTC + Tamper
- 5 WKUP pins

250 us wake-up

STM32L4 Power Mode

Shutdown Mode

Available Peripheral

- GPIO
- DMA
- FSMC
- QSPI
- BOR
- PVD, PVM
- LCD
- USB OTG
- USART
- LP UART
- I2C 1 / I2C 2
- I2C 3
- SPI
- CAN
- SDMMC
- SWPMI
- SAI
- DFSDM
- ADC
- DAC
- OPAMP
- COMP
- Temp Sensor
- Timers
- LPTIM 1
- LPTIM 2
- IWDG
- WWDG
- Systick Timer
- Touch Sens
- RNG
- AES
- CRC

I/Os can be configured
w/ or w/o pull-up
w/ or w/o pull-down
But floating when exit from Shutdown

Shutdown → **64 nA @ 3.0V**
30 nA @ 1.8V

Zzz

- Cortex M4
- Flash (1MB)
- SRAM 1 (96KB)
- SRAM 2 (32KB)

- Main regulator (MR)
- Low Power regulator (LPR)

Available Clock

- HSI
- HSE
- LSI
- **LSE**
- MSI

- **Backup domain**
  - **Backup Register (32x32-bits)**
  - **RTC**

Wake-up event

- **NRST**
- **RTC + Tamper**
- **5 WKUP pins**

256 us wake-up

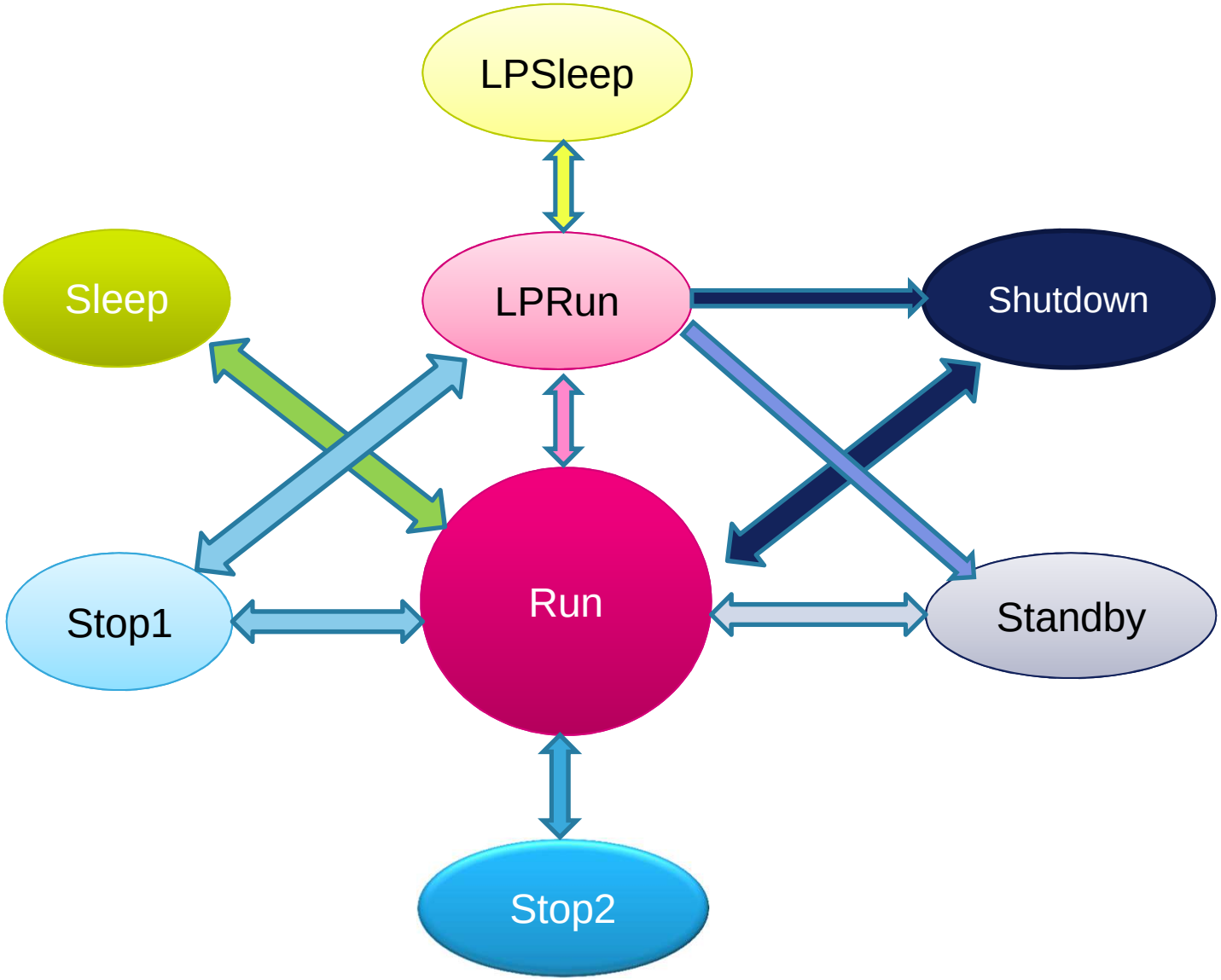
Low-power modes summary

Mode	Regulator	CPU	Flash	SRAM	Clocks	Peripherals In Bold : wakeup source	Consumption @ 1.8V	Wakeup time
Run	R1	Yes	ON(1)	ON	Any	All	131 µA/MHz	N/A
	R2					All except OTG, SDMMC, RNG	112 µA/MHz	
LPRun	LPR	Yes	ON(1)	ON	Any *except PLL*	All except OTG, SDMMC, RNG	135 µA/MHz	TBD
Sleep	R1	No	ON(1)	ON(2)	Any	All **Any IT or event**	37 µA/MHz	6 cycles
	R2						35 µA/MHz	
LPSleep	LPR	No	ON(1)	ON(2)	Any *except PLL*	All except OTG, SDMMC, RNG **Any IT or event**	40 µA/MHz	6 cycles
Stop 1	LPR	No	OFF	ON	LSE/LSI	**Reset pin, all I/Os BOR,PVD,PVM,RTC,LCD,IWDG, COMPx**,DACx,OPAMPx,**USARTx, LPUART,I2Cx,LPTIMx,OTG_FS, SWPMI**	6.6µA w/o RTC 6.9 µA w/RTC	4 µs RAM 6 µs Flash
Stop 2	LPR	No	OFF	ON	LSE/LSI	**Reset pin, all I/Os BOR,PVD,PVM,RTC,LCD,IWDG, COMPx,LPUART,I2C3,LPTIM1**	1.2 µA w/o RTC 1.4 µA w/RTC	5 µs RAM 7 µs Flash
Standby	LPR	DOWN	OFF	SRAM2 ON	LSE/LSI	**Reset pin, 5 WKUPx pins BOR, RTC, IWDG**	+ 235 nA	14 µs
	OFF			DOWN			128 nA w/o RTC 433 nA w/RTC	
Shutdown	OFF	DOWN	OFF	DOWN	LSE	**Reset pin, 5 WKUPx pins RTC**	30 nA w/o RTC 329 nA w/RTC	256 µs

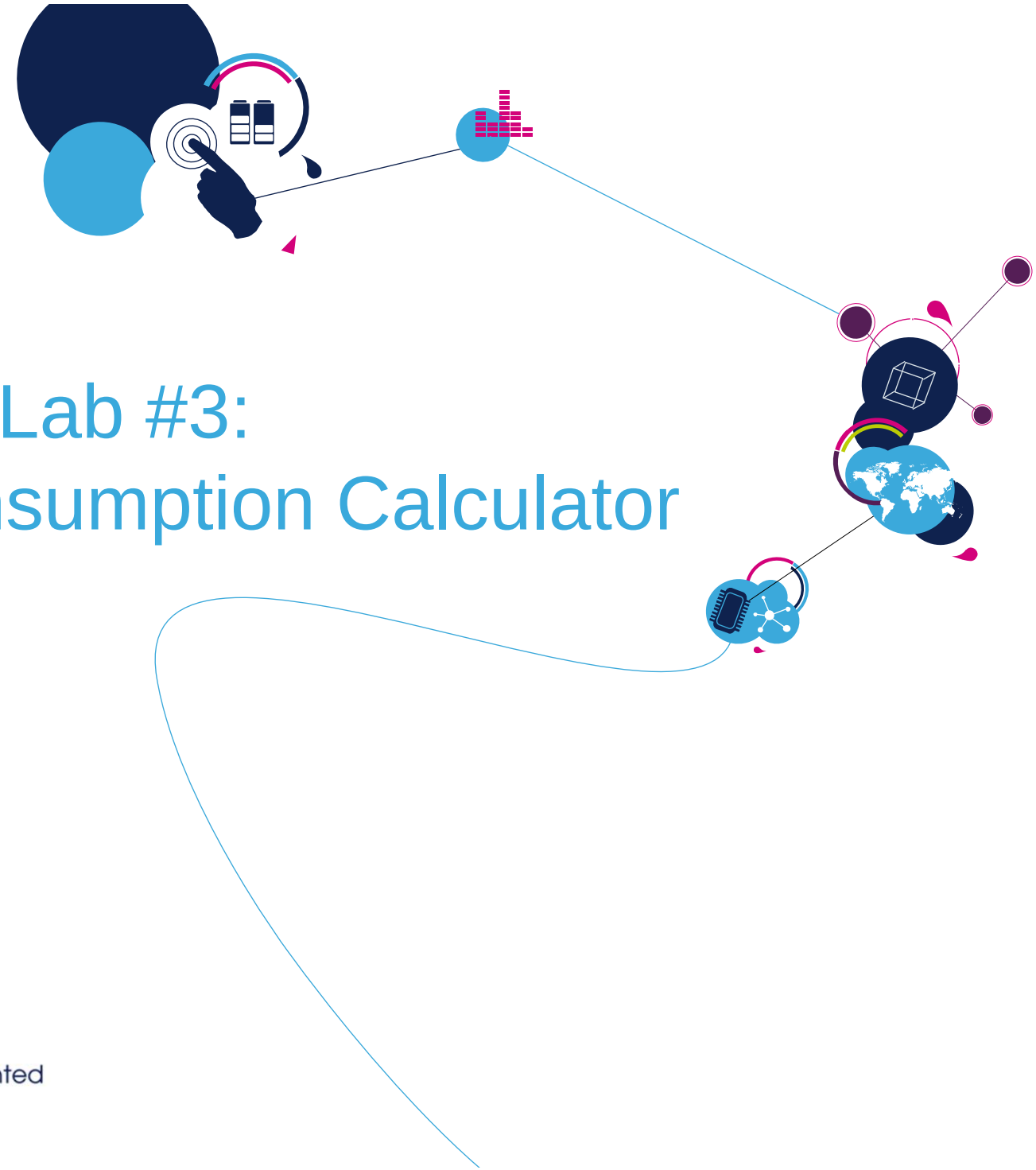
1. Can be put in power-down and clock can be gated off
2. SRAM1 and SRAM2 can be gated off independently

life.augmented

Low-power modes transitions



life.augmented

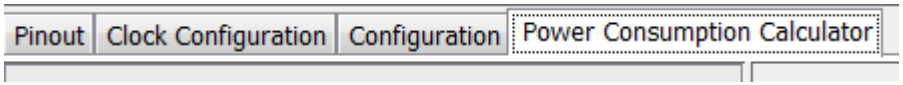


Hands-On Lab #3:
Power Consumption Calculator

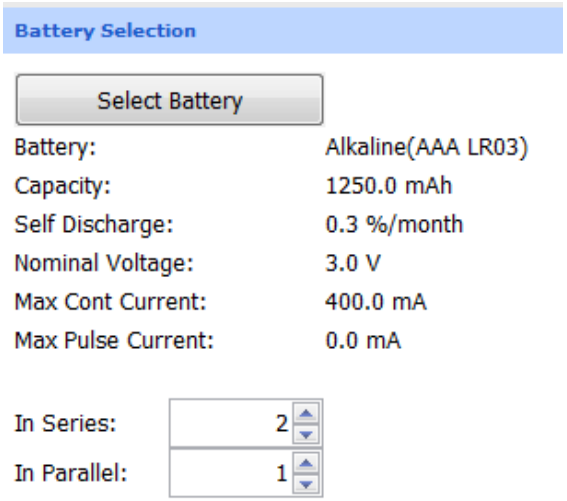
life.augmented

Power Consumption Calculator Lab

- **Click on the Power Consumption Calculator tab in CubeMX**



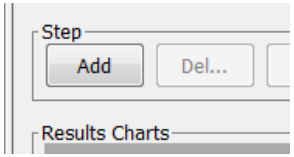
- **Select two AAA Alkaline batteries (in series) as our power source:**



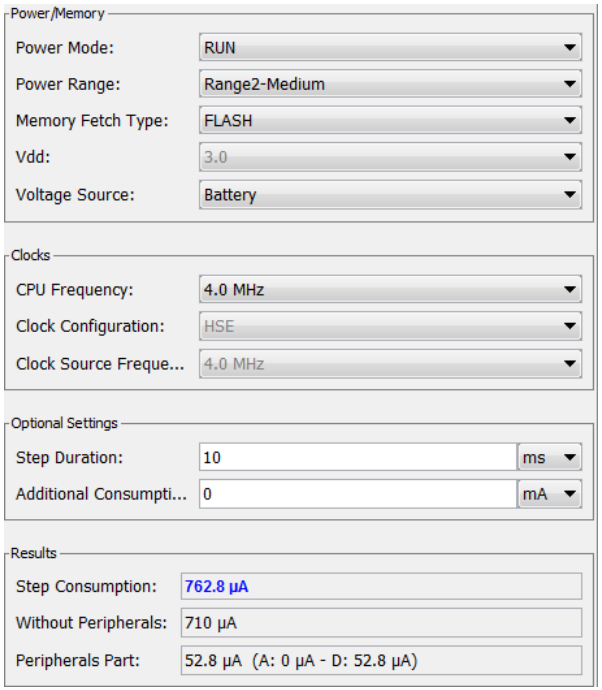
life.augmented

Power Consumption Calculator Lab

- **Add a step to our power sequence: Click: Step.. Add**



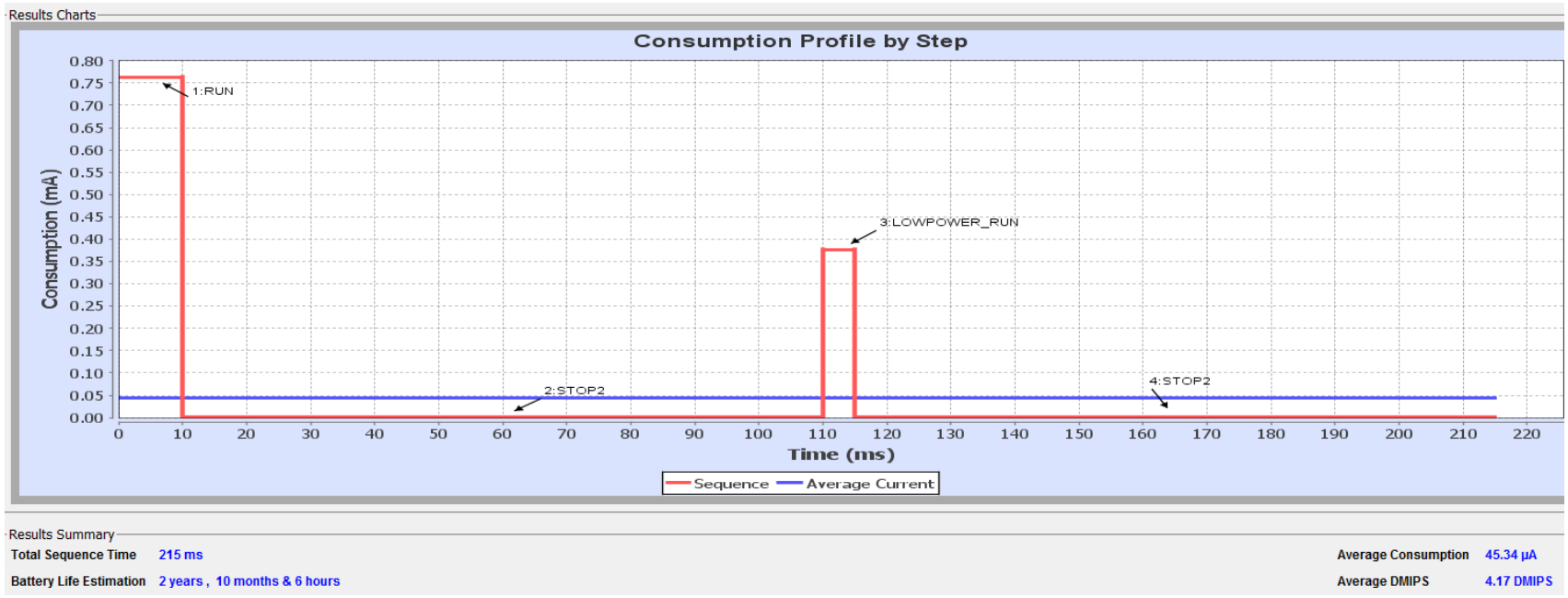
- **Configure a 10ms step:**
  - 4.0MHz Range2 RUN mode
  - GPIOB, GPIOE and USART2 peripherals enabled
  - Click “Add”
  - Resulting step consumption should be 762.8uA



life.augmented

Power Consumption Calculator Lab

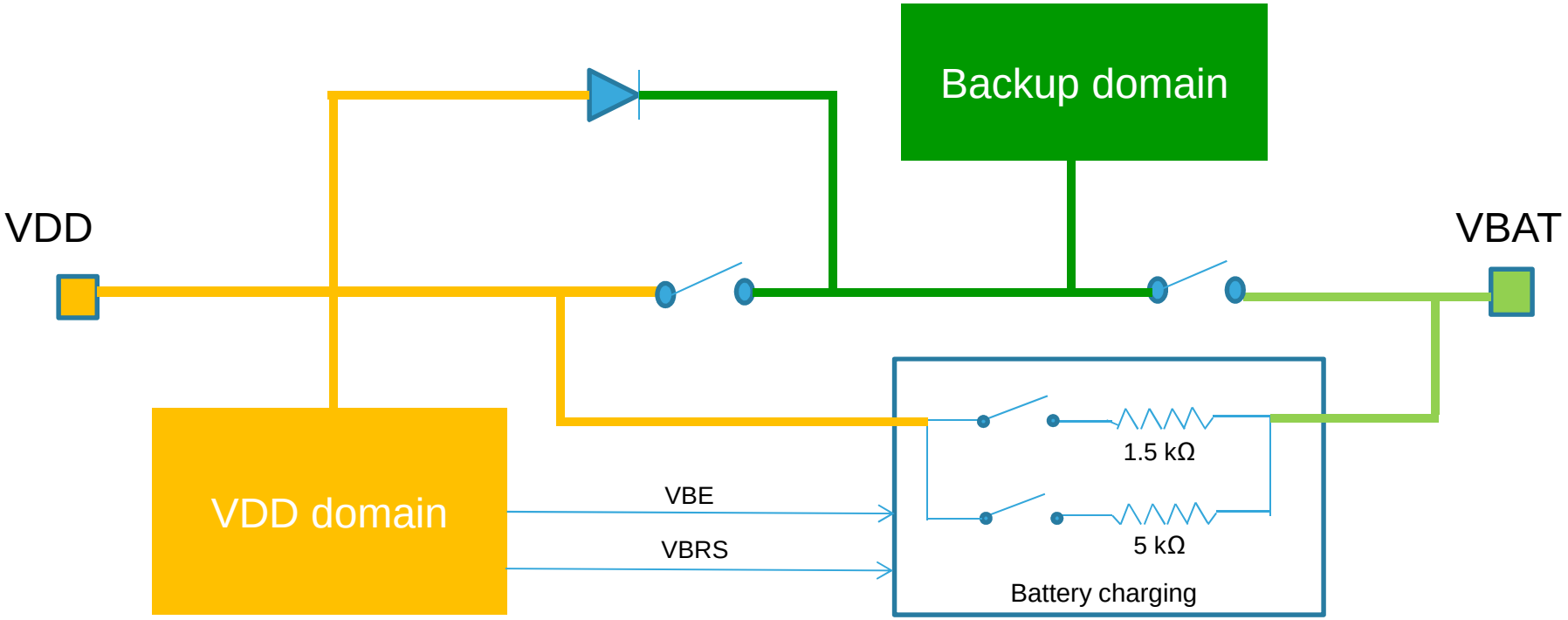
- **Add three more additional power steps:**
  - **STOP2 mode, Battery power source, ALL Clocks OFF, 100ms duration**
  - **LPRUN mode, FLASH fetch type, Battery power source, 2.0MHz freq, 5ms duration, GPIOB, GPIOE active**
  - **Duplicate Step #2 (STOP2 mode) using the DUPLICATE button**
- **Check the result:**



life.augmented

VBAT backup domain

- VBAT charging : allows charging a super-cap on VBAT through internal resistor when VDD is present
  - Battery charging is enabled by setting **VBE** bit in the PWR_CR4 register.
  - **VBRS** bit value in the PWR_CR4 register selects the resistor value



life.augmented

STM32L4
Peripheral details 1/2

life.augmented

Smart peripherals
Metering

Digital **F**ilter for **S**igma **D**elta **M**odulators
8 x parallel inputs
with up to 24-bit data
output resolution

LCD Display
88×40 or 4×44
with step-up converter

V_{BAT} with RTC
for battery backup
240 nA in V_{BAT} mode
for RTC and
32x 32-bit backup registers

STM32L4

Anti Tamper pin
3 x tamper pins
for battery domain

TRNG & AES
for Security
128-/256-bit AES
key encryption hardware
accelerator

Electricity/Gas /Water Smart Meter

SPI / UART/ SDIO for Wireless
3x SPIs (4x SPIs with the Quad SPI)
6x USARTs (ISO 7816, LIN, IrDA, modem)
1 x SDIO

FSMC
External memory interface
for static memories supporting SRAM,
PSRAM, NOR and NAND

I/Os Up to 114 fast I/Os for buttons & relays

Smart Peripherals
Industrial Sensors

Motor Control :
2x 16-bit advanced
motor-control timers
3x 12-bit ADCs: 5 MSPS,
with up to 16-bit with hardware
oversampling, 200 µA/MSPS

LCD Display
8×40 or 4×44
with step-up converter

CAN Bus
(2.0B Active)

High temperature
from -40°C
up to + **125°C**

TRNG & AES
for Security
128/256-bit AES
key encryption hardware accelerator

STM32L4

SPI / UART
3x SPIs (4x SPIs with the
Quad SPI)
6x USARTs (ISO 7816, LIN,
IrDA, modem)

FSMC
External memory interface
for static memories supporting
SRAM, PSRAM, NOR and NAND

I/Os
Up to 114 GPIOs

I²C
3x I²C FM+(1 Mbit/s), SMBus/PMBus

Smart peripherals

Fitness tracker application

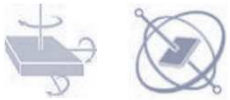
Digital **F**ilter for **S**igma **D**elta **M**odulators
with PDM microphone input support

TFT Display

FSMC
Parallel interface to TFT
SPI
Up to 40 MHz speed

STM32L4

Sensors



I²C
3x I²C FM+

Batch **A**cquisition **M**ode (BAM)

USB
USB OTG 2.0
full-speed,
LPM and BCD

SPI / UART
3x SPIs
Quad SPI
6x USARTs

OPAMP
2x with built-in PGA

DAC
Low-power sample and hold

ADC
3× 12-bit ADC 5 MSPS,

SAI
2x serial audio interfaces

SWP
Single wire protocol
master interface (SWPMI)

life.augmented

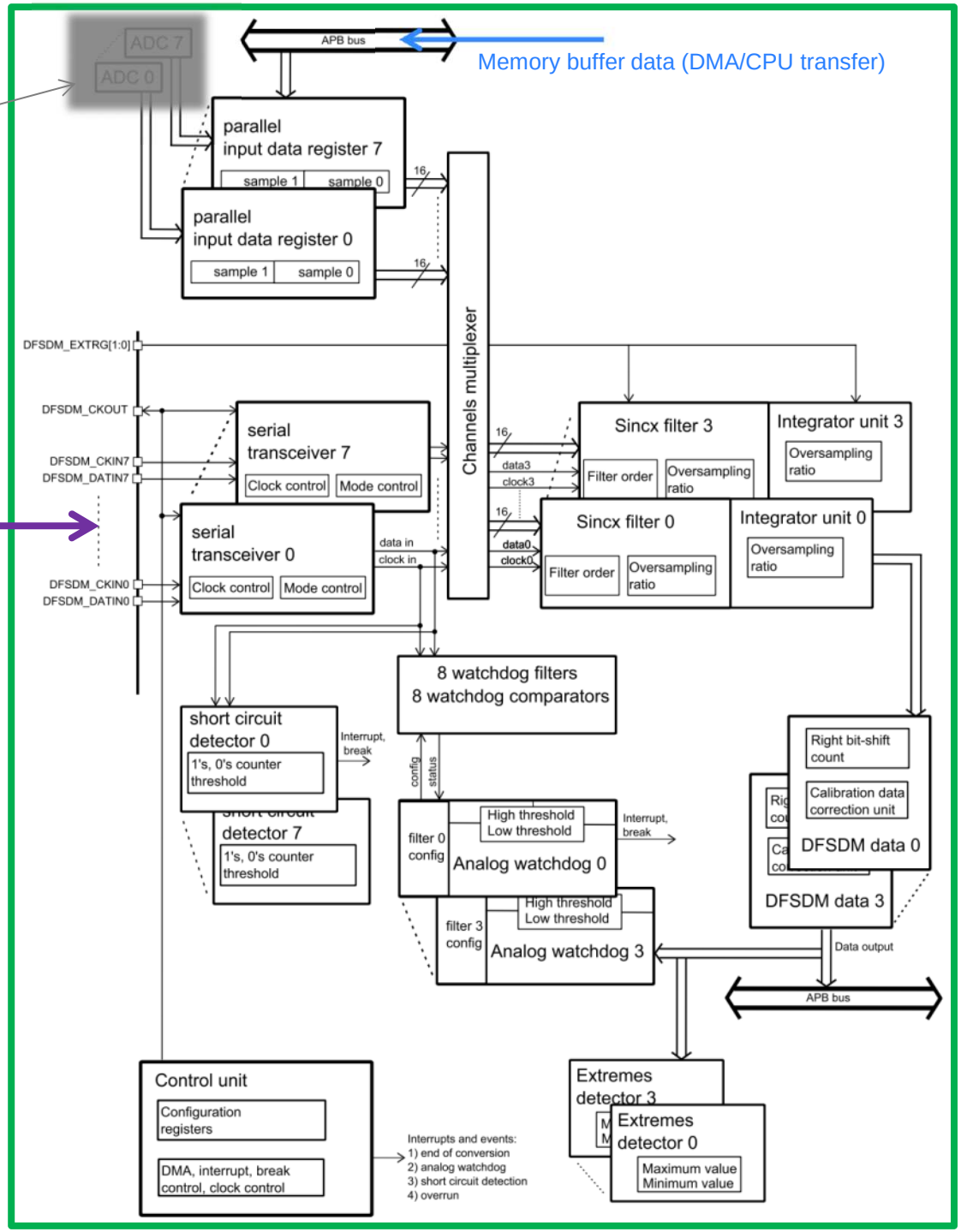
DFSDM Introduction

- External ΣΔ modulators on market:
  - This is external standalone device: ADC converter on sigma delta principle
  - Analog input (usually differential) and digital output
    - Precision: ~16-bit resolution
  - Provides digital output as fast 1-bit data stream => serial interface
    - Up to 20MHz speed of serial data
  - Wide range of suppliers (ST and others)

- STM32 interface: DFSDM = **D**igital **F**ilter for **S**igma **D**elta **M**odulators

  Implements complete post-processing from external ΣΔ modulators outputs:
  - Receiving of data streams from ΣΔ modulators (in various serial data formats)
  - Digital filtering of data stream (final 24-bit result)
  - Security/emergency functions

Block diagram



life.augmented

Decode, Filter, Average, Process

- Up to 8 Serial transceivers:
  - Receive and decode raw serial bitstreams, providing data/clock to filter stage
  - 1-wire Manchester coded mode or SPI clock/data
- $Sinc^x$ filter performs input stream digital filtering:
  - $Sinc^1$, $Sinc^2$, $Sinc^3$, $Sinc^4$, $Sinc^5$, FastSinc, No filter
  - Programmable $Sinc^x$ oversampling ratio (1 to 1024 filter samples)
  - Output filter resolution is 31-bit max
- Integrator stage performs data averaging from digital filter, 1-256 samples
- Post processing:
  - Offset compensation
  - Programmable right bit shifting for data formatting
- Additional functions
  - Min/Max extremes detection
  - Analog watchdog (to watch for final data boundaries overflow)
  - Break signal generation

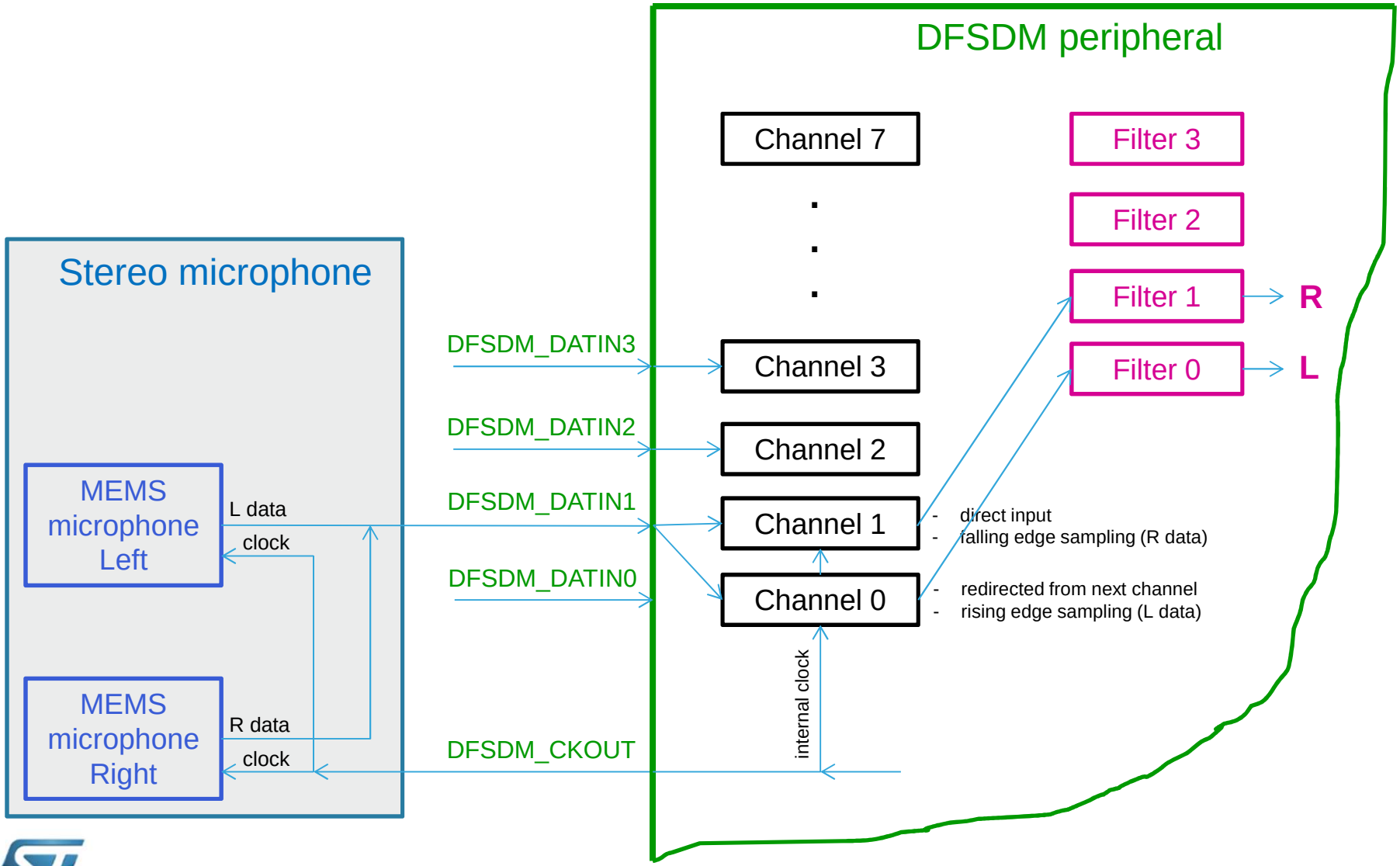
MEMS microphone support (PDM)

- Function
  - MEMS microphone provides pulse density modulated (PDM) data signal – like the ΣΔ modulator.
  - PDM microphone has stereo support (if two connected in parallel):
    - Rising clock = Left audio data, Falling clock = Right audio data

- Implementation into DFSDM transceivers
  - Channels data (left vs. right) are separated inside:
    - Each DFSDM channel transceiver inputs can be redirected to next channel inputs
    - Configuration of those 2 channels differs only in active sampling edge
  - Clock signal provided by DFSDM_CKOUT – PDM microphones are slaves

life.augmented

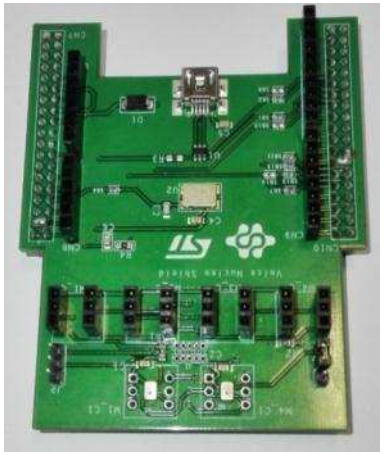
MEMS microphone application - schematic



life.augmented

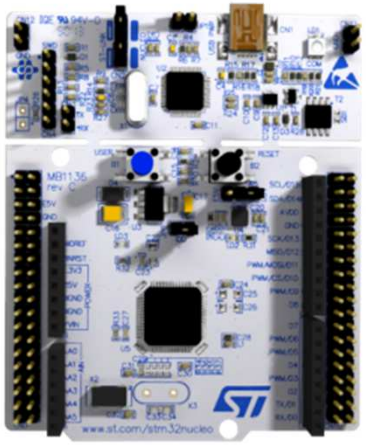
Voice recognition Demonstration

- STM32L4 with voice recognition algorithm controls an Android remote device thanks to Bluetooth Low Energy communication



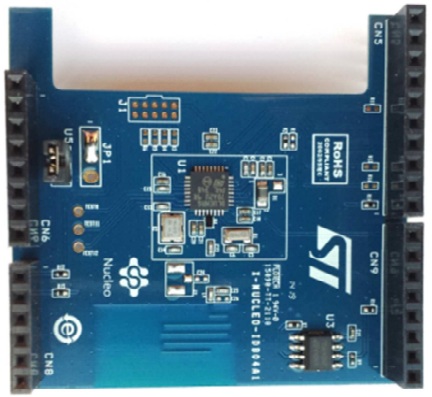
Microphone Shield With Digitals MEMS (MP34DT01)

+



Nucleo STM32L4

+



ST BlueNRG BLE RF Arduino Shield

+

CR2032

Voice recognition function blocks

Always on acquisition
(PDM to PCM & signal conditioning)

Voice activity detector

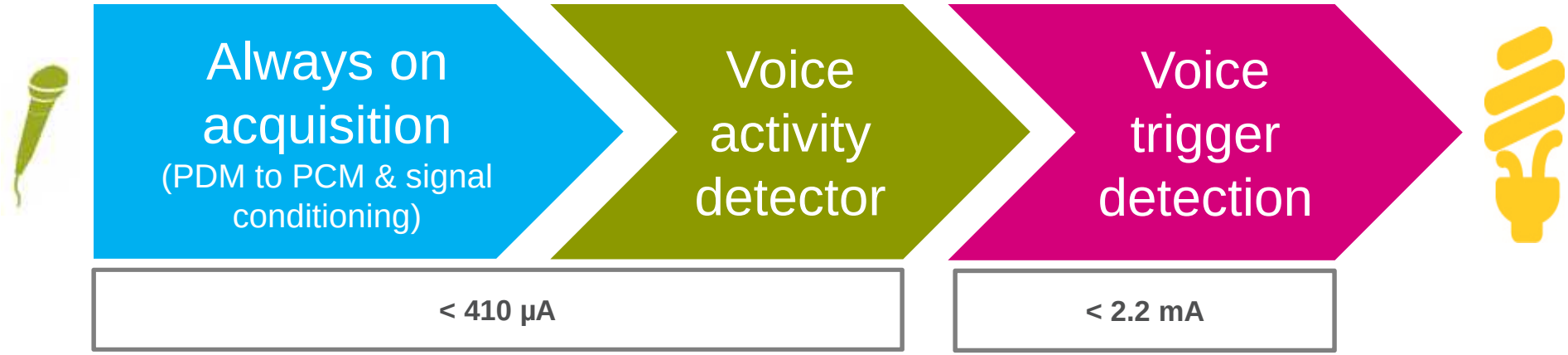
Voice trigger detection

Always on BLE connection

life.augmented

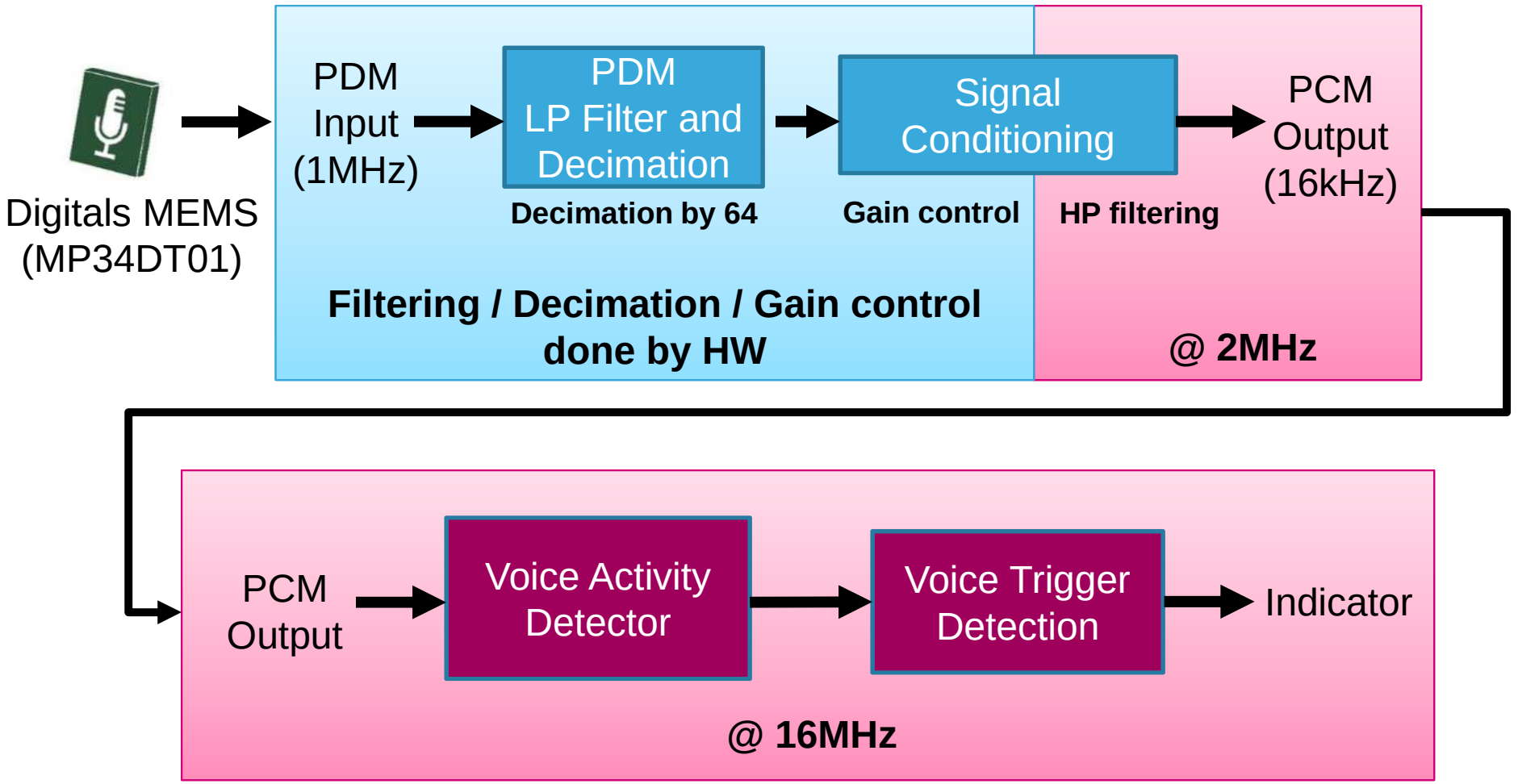
Low power audio DSP replacement

Voice recognition example



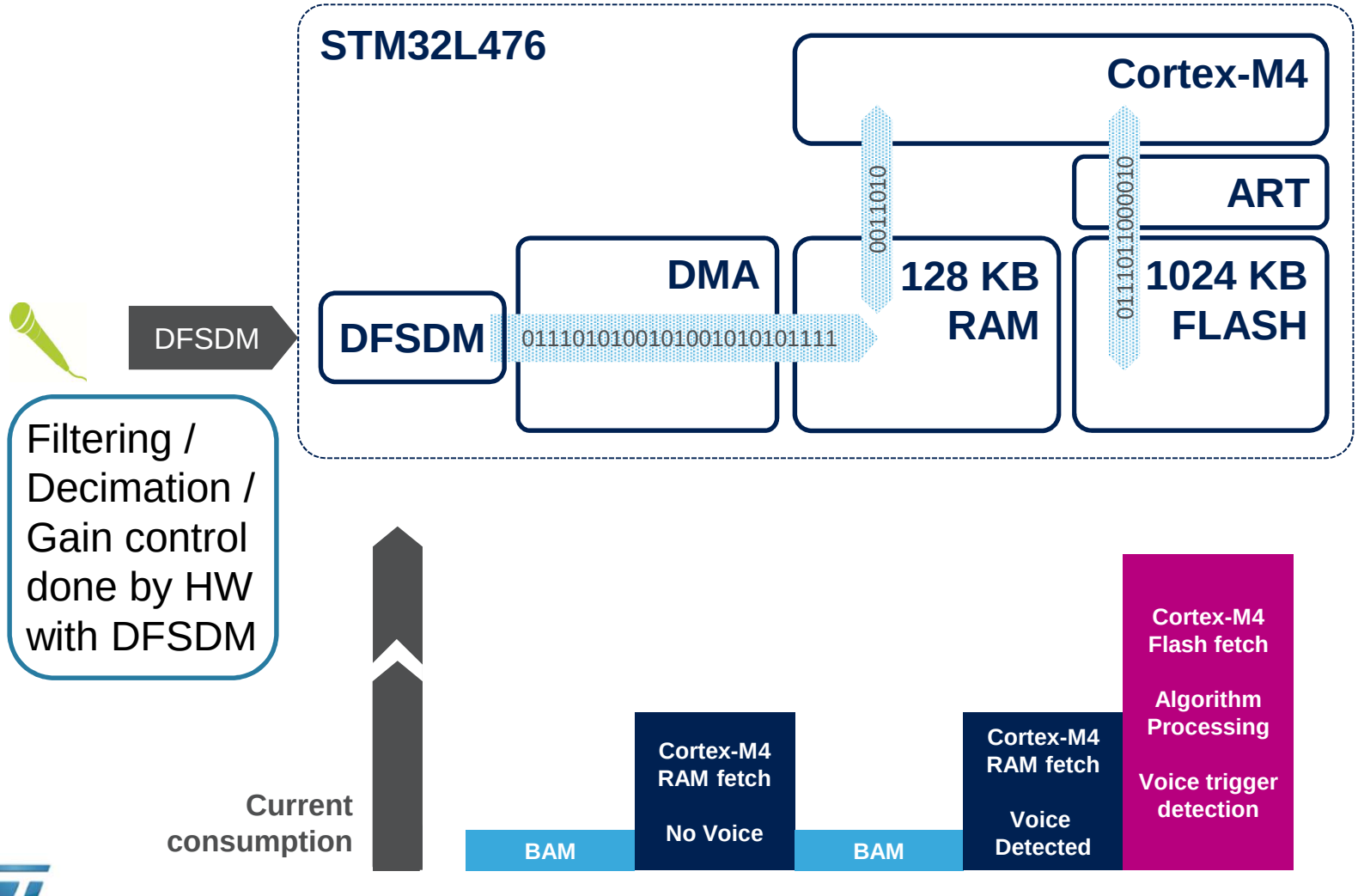
Sub-microwatt acquisition thanks to PDM to PCM HW processing with DFSDM and low power Batch Acquisition Mode (BAM)

Hardware/Software blocks



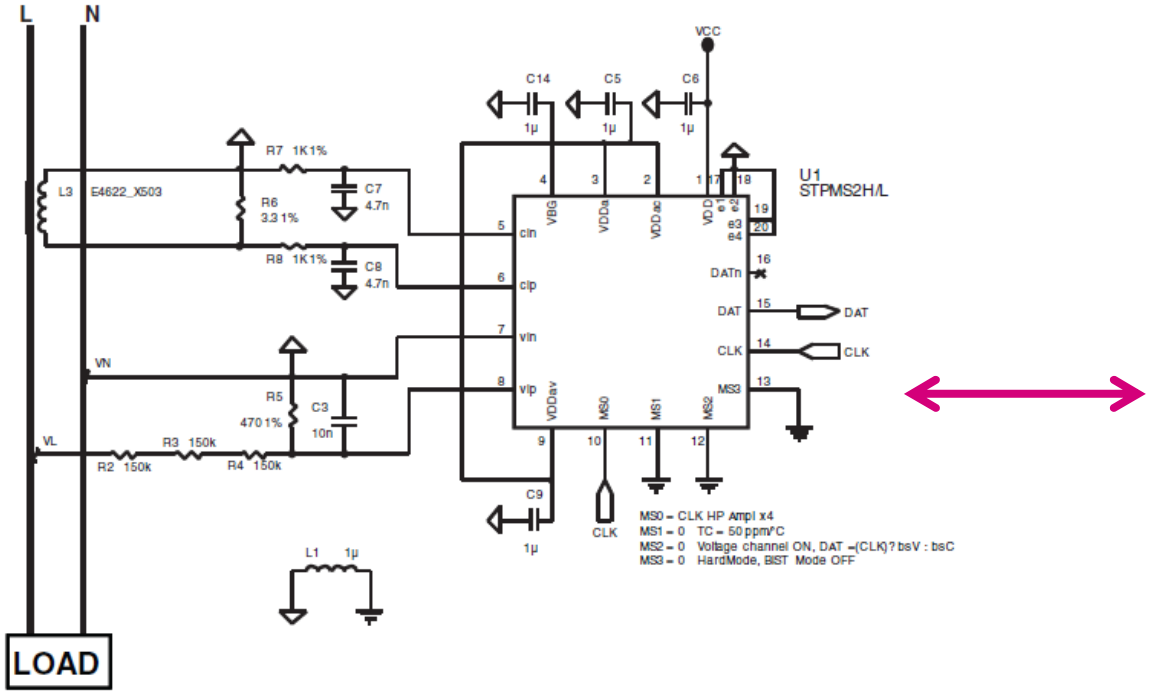
BAM explained

Voice recognition use case



Smart peripherals - DFSDM

Metering



STM32L4

Electricity Meter

STPMS2

Smart sensor II dual-channel 1-bit, 4 MHz, second-order sigma-delta modulator with embedded PGLNA

Features

- V_{CC} supply range 3.2 V - 5.5 V
- Two second-order sigma-delta (ΣΔ) modulators
- Programmable chopper-stabilized low noise and low offset amplifier
- Supports 50-60 Hz, EN 50470-1, EN 50470-3, IEC 62053-21, IEC 62053-22 and IEC 62053-23 standards specs for class 1, class 0.5 and class 0.2 AC watt meters
- STPM02H: less than 0.5% error over 1:10000 range
- STPM02L: less than 0.5% error over 1:5000 range
- Precision voltage reference: 1.23 V with programmable TC (STPMS2L only)
- Internal low drop regulator @ 3 V (typ.)

Applications

- Power metering
- Motor control
- Industrial process control
- Weight scales
- Pressure transducers

QFN16 (4 x 4)

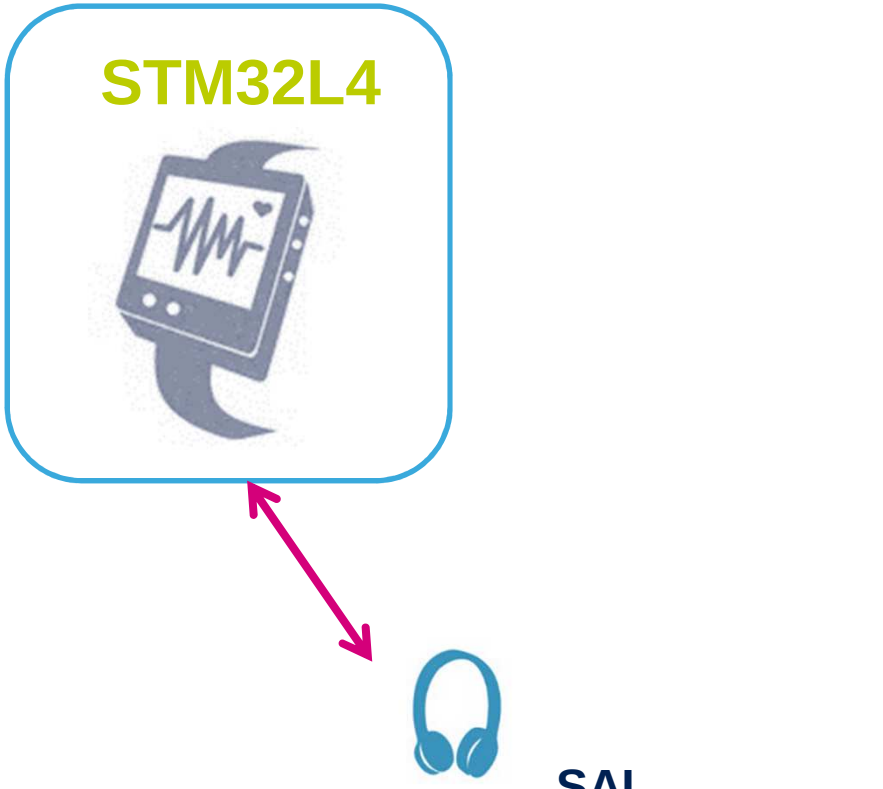
or multi-phase energy meters along with the STPMC1 device, a digital signal processor designed for energy measurement. This device can be used in medium and high resolution measurement applications where single or double inputs must be monitored at the same time. The STPMS2 are mixed signal ICs consisting of an analog and digital section. The analog section consists of a programmable gain, low noise choppered amplifier, two second-order ΔΣ modulator blocks, a band-gap voltage reference, a low-drop voltage regulator and DC buffers, while the digital section consists of a clock generator and output multiplexer.

life.augmented

Smart peripherals – Serial Audio Interface
Fitness tracker application

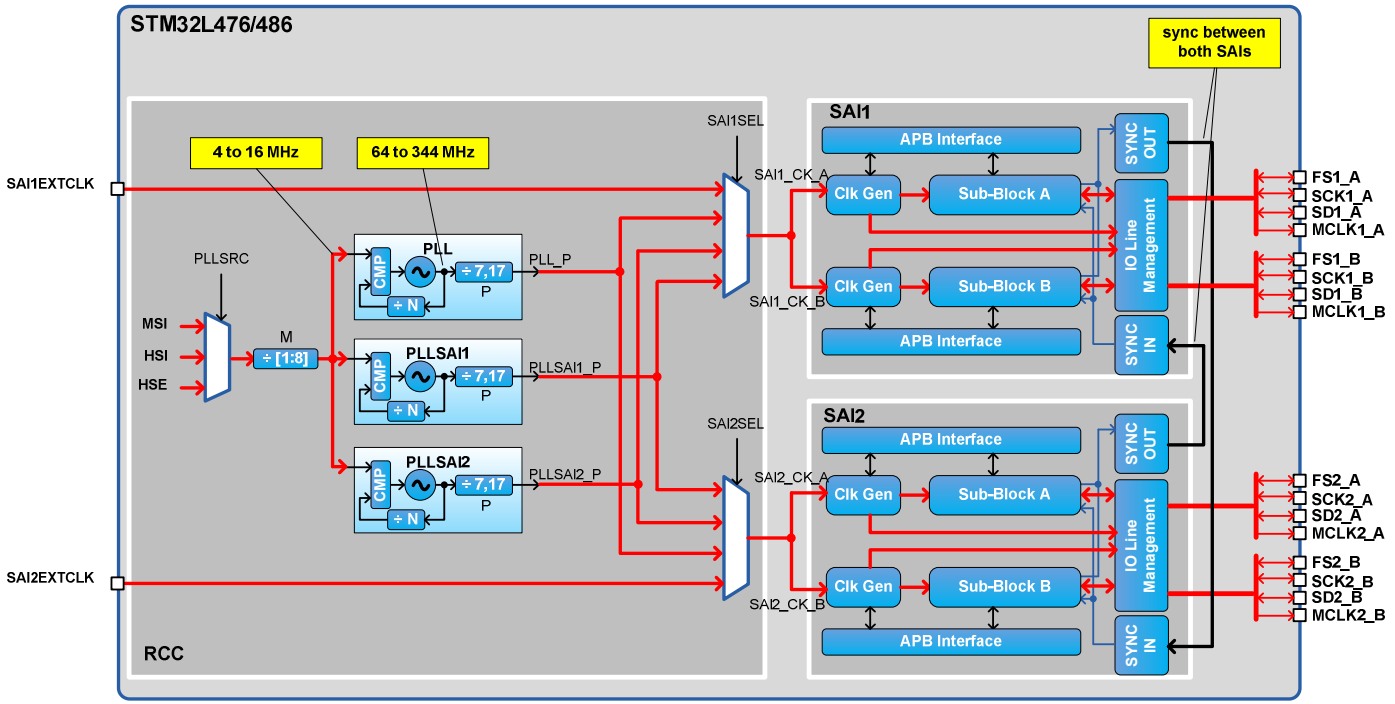
Serial Audio Interface supports a wide set of audio protocols thanks to its flexible architecture:

- **I2S Philips standards,** (**I**nter-**I**C **S**ound)
- **I2S LSB or MSB-justified,** (I2S variant)
- **SPDIF Output,** (**S**ony/**P**hilips **D**igital **I**nter**F**ace)
- **PCM,** (**P**ulse **C**ode **M**odulation)
- **TDM,** (**T**ime **D**ivision **M**ultiplexing)
- **AC'97,** (**A**udio **C**odec **'97** from Intel)



SAI Features

- Two independent audio sub-blocks:
  - Transmitter and/or receiver, Master or Slave
  - Synchronous or asynchronous modes between the audio sub-blocks
  - Clock generator for each audio sub-block to target independent audio frequencies
- 8-word integrated FIFOs for each sub-block, up to 16 slots available
- Mute mode, Stereo/Mono and companding mode supporting µ-Law and A-Law.
- Configurable LSB/MSB, data/slot sizes, sampling edges, # of bits, frame shape, etc..



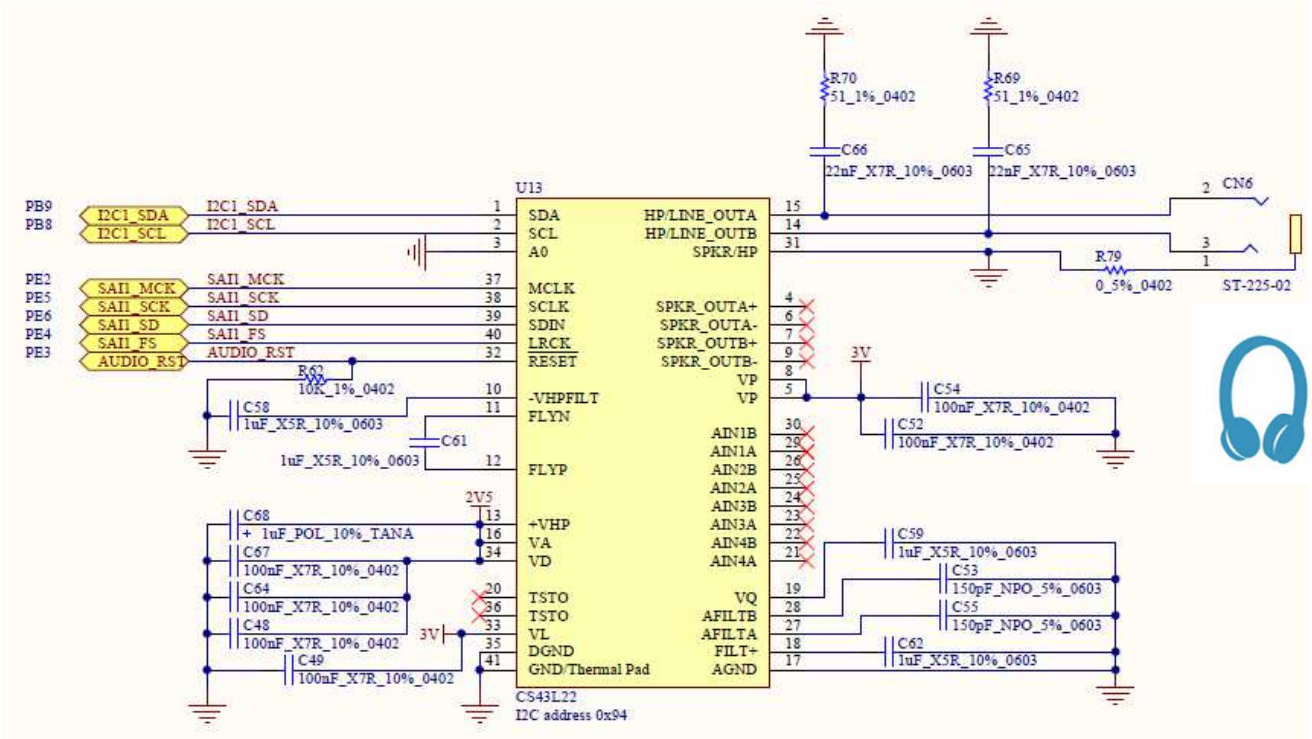
life.augmented

Smart peripherals

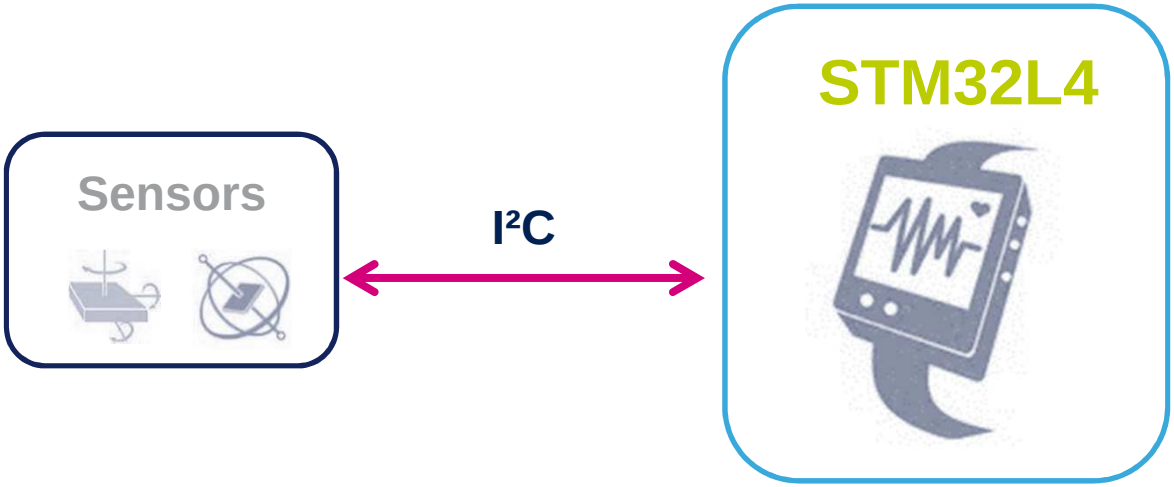
SAI: Discovery Board Implementation:

STM32L4

SAI in I2S Philips mode



Smart peripherals – I²C
Fitness tracker application

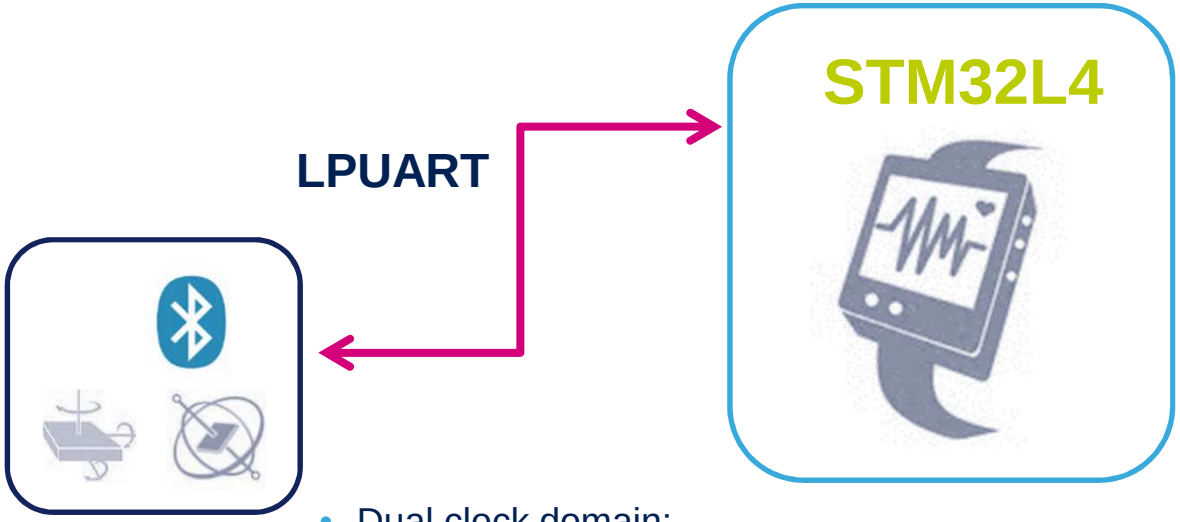


- All known I²C limitations are corrected!
  - Flag management improvements (L0/F3-like) for easy use
  - Programmable filter on input pins (analog & digital filters)
- Wake-up on address matches – STOP1 (I²C1/2/3), STOP2 (I²C3)
- Fast mode Plus (up to 1Mbits/s) + 20mA
- Dual clock domain allows high baudrates and low CPU clocks to save power

life.augmented

Smart peripherals – USART/LPUART

Fitness tracker application



- Dual clock domain:
  - UART functionality and wake up from Stop mode (Stop 1 and Stop 2)
  - Baudrate programming independent of PCLK
  - Four clock sources: **LSE** (32.768KHz), **HSI**, **PCLK**, **System Clock**
  - Supports up to **9600** baud via **32.768KHz** LSE!
  - To wakeup from STOP mode, clock source must be LSE or HSI
- 3 Wakeup events: Address match, START bit detection or RXNE (receive buffer not empty)
- **KEY message: LPUART adds STOP2 wakeup, but is a subset of USART, however it is much lower power**

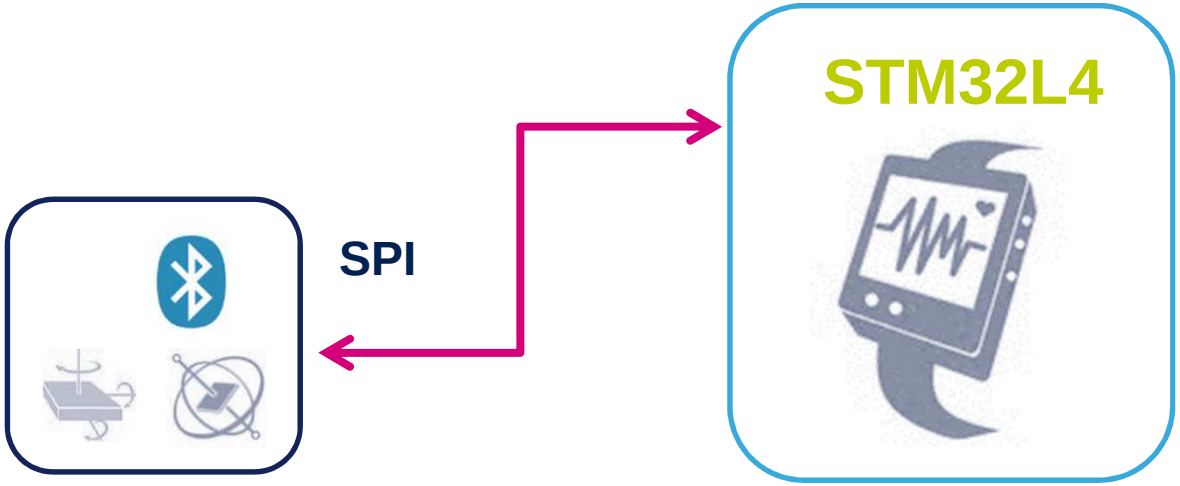
STM32L4 USART Implementation

- 3x USART's, 2x UART's and LPUART:

USART features	USART1/2/3	UART4/5	LPUART
Hardware Flow Control	YES	YES	YES
Continous communication using DMA	YES	YES	YES
Multiprocessor communication	YES	YES	YES
Synchronous mode	YES	NO	NO
Smartcard mode	YES	NO	NO
Single wire half duplex mode	YES	YES	YES
IrDA	YES	YES	NO
LIN	YES	YES	NO
Dual clock / wake up from STOP1	YES	YES	YES
Dual clock / wake up from STOP2	NO	NO	YES!!
Receiver timeout	YES	YES	NO
Modbus Communication	YES	YES	NO
Autobaudrate detection	YES	YES	NO
RS-485 Driver enable	YES	YES	YES

life.augmented

Smart peripherals - SPI



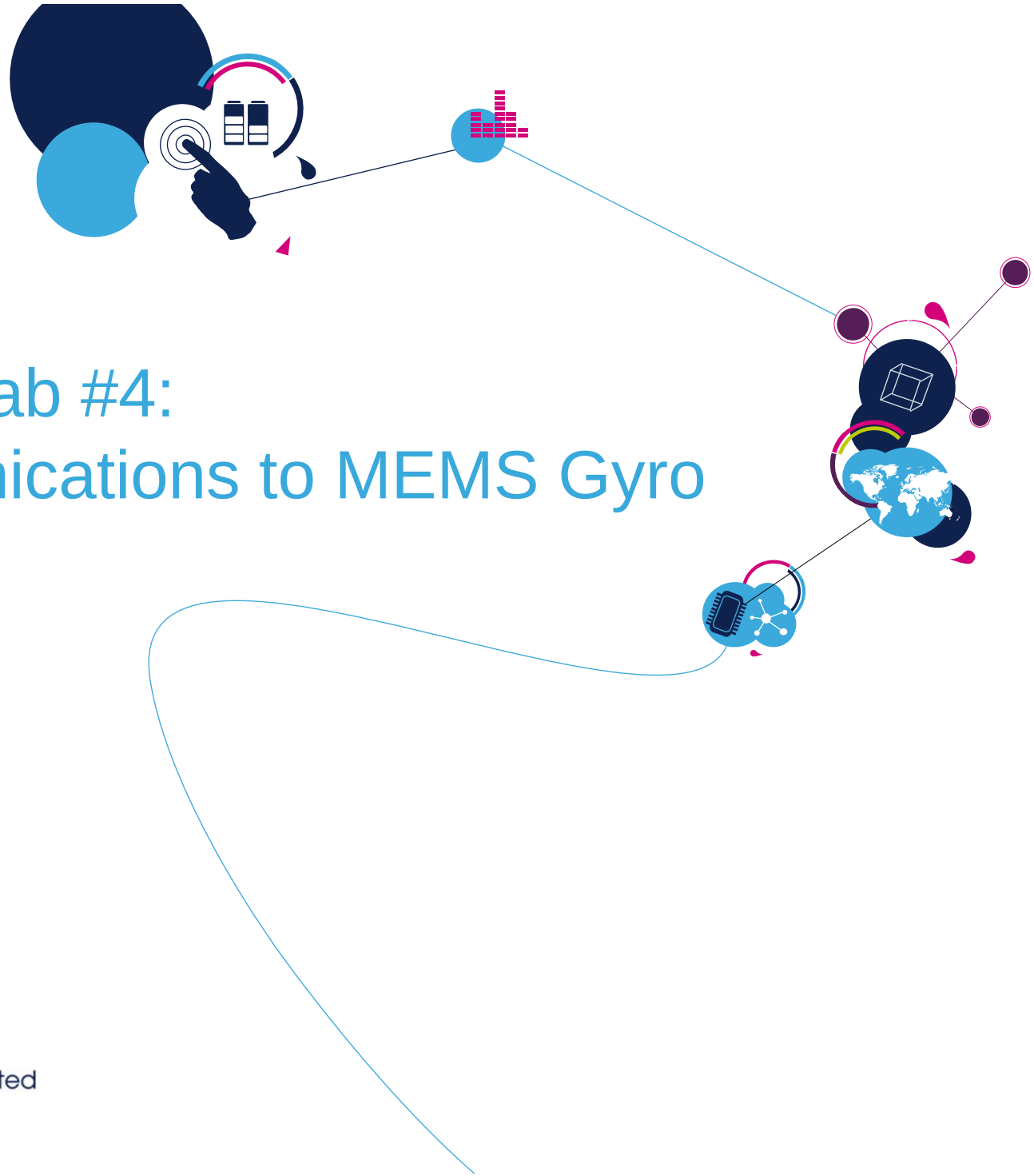
- Programmable frame from 4 to 16bit (bit granularity)
- 32-bit FIFO to optimize bus bandwidth (performance & power consumption)
- BUSY status bit IP fix
- Up to 40 Mbits/s in master mode
- I^2S functionality no longer supported in SPI. Now in SAI peripheral

Smart peripherals
Sensor Hub scenario

Batch **A**cquisition **M**ode (BAM)

Optimized mode for transferring data while the system is in low power mode

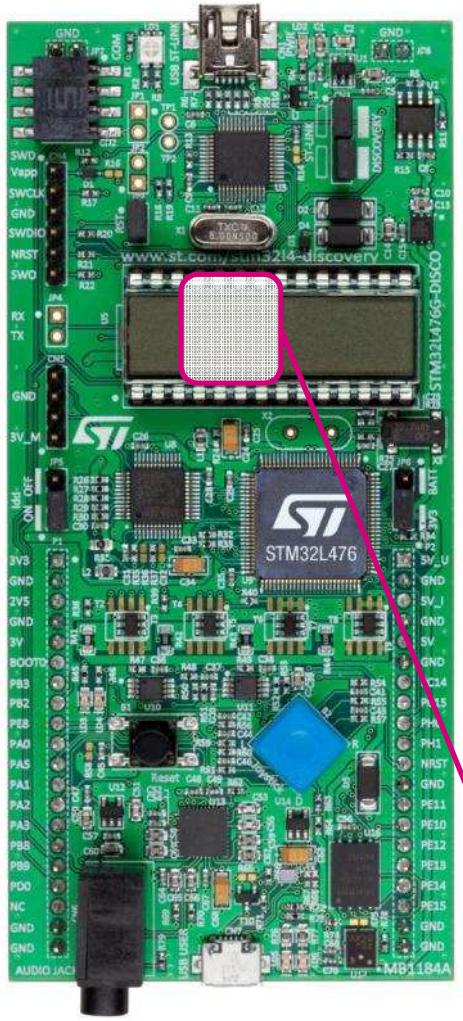
- Only the needed communication peripheral + 1 DMA + 1 SRAM (SRAM1 or SRAM2) are configured with clock enable in Sleep mode
- Flash is in power-down mode
- Enter either Sleep or Low-power sleep mode
  - I2C clock can be at 16 MHz, allowing FM+ support



Hands-On Lab #4:
SPI communications to MEMS Gyro

life.augmented

Hands-On Lab #4 – SPI communications to 3-axis Gyro



Adding to the existing CubeMX project, we will add SPI communications to the L3GD20 MEMS gyroscope:

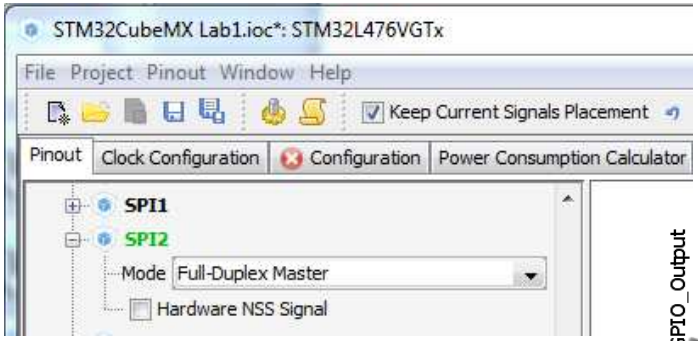
Set up additional GPIO / Clocks:
PD1 – SPI2 SCK – Alt. Fn. Push/Pull
PD3 – SPI2 MISO – Alt. Fn. Push/Pull
PD4 – SPI2 MOSI – Alt. Fn. Push/Pull
PD7 – nCS – GPIO Output, Push/Pull mode

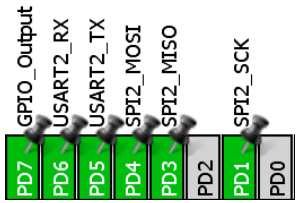
SPI2 settings:
Full-Duplex Master
Motorola, 8-bit MSB-first
Prescaler /16
CPOL = Low, CPHA = 1-Edge
No CRC
Software NSS

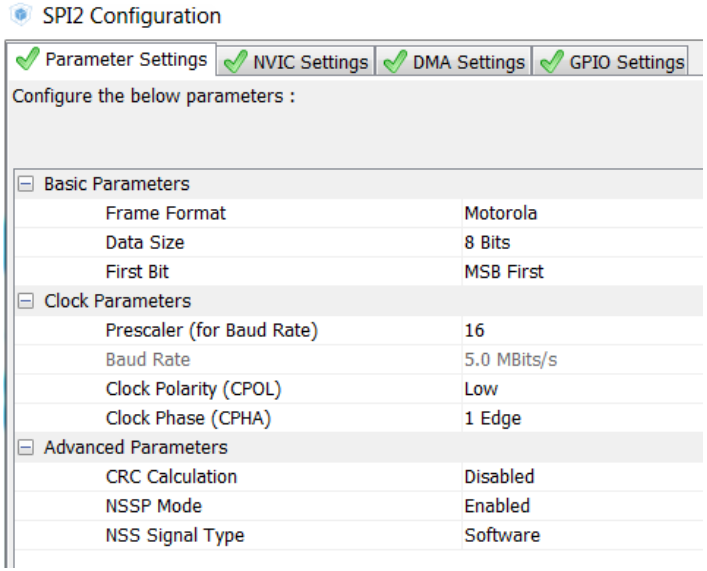
User Code HAL function calls required to be added to main.c:
C:\STM32L4Seminar\Labs**lab4_spi_gyro.c**

Gyro (under LCD) data is routed to SPI2

Regenerate Source Code for Lab1





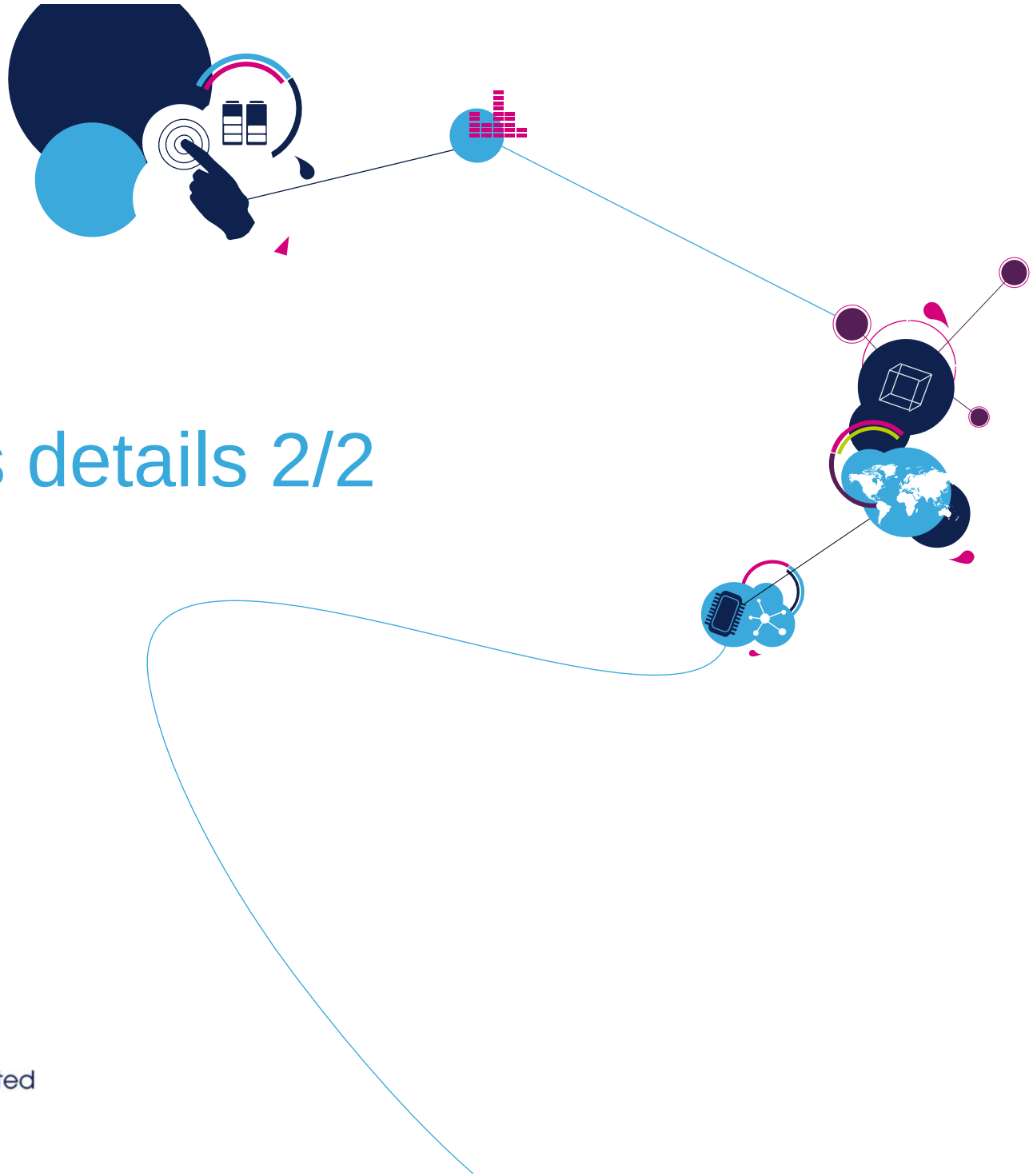


Pin Name	Signal on Pin	GPIO mode	GPIO Pull-up/Pull-down	Maximum output speed	Fast Mode	User Label	Modified
PD1	SPI2_SCK	Alternate Function Push Pull	No pull-up and no pull-do...	High	n/a	SPI_Clock	☑
PD3	SPI2_MISO	Alternate Function Push Pull	No pull-up and no pull-do...	High	n/a	SPI_In	☑
PD4	SPI2_MOSI	Alternate Function Push Pull	No pull-up and no pull-do...	High	n/a	SPI_Out	☑

- Needed code bits from C:\STM32L4Seminar\Labs**lab4_spi_gyro.c** into main.c:
  - **Add code bits between:** **/* USER CODE BEGIN PV */** **/* USER CODE END PV */**
  - **Add code bits between:** **/* USER CODE BEGIN PFP */** **/* USER CODE END PFP */**
  - **Add code bits between:** **/* USER CODE BEGIN 2 */** **/* USER CODE END 2 */**
  - **Add code bits between:** **/* USER CODE BEGIN WHILE */** **/* USER CODE END WHILE */**
  - **Add code bits between:** **/* USER CODE BEGIN 4 */** **/* USER CODE END 4 */**

- Open a terminal emulator, using USART2 settings, Virtual COM port xx
- Rebuild, Program/Debug, Run!

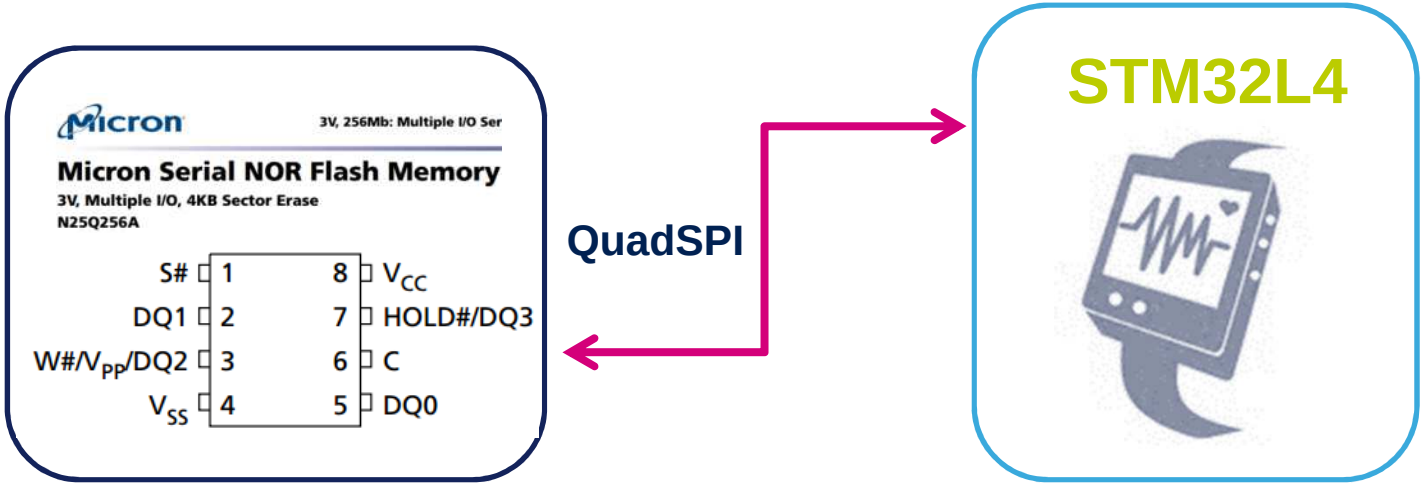
life.augmented



STM32L4
Peripherals details 2/2

life.augmented

Smart peripherals - QuadSPI

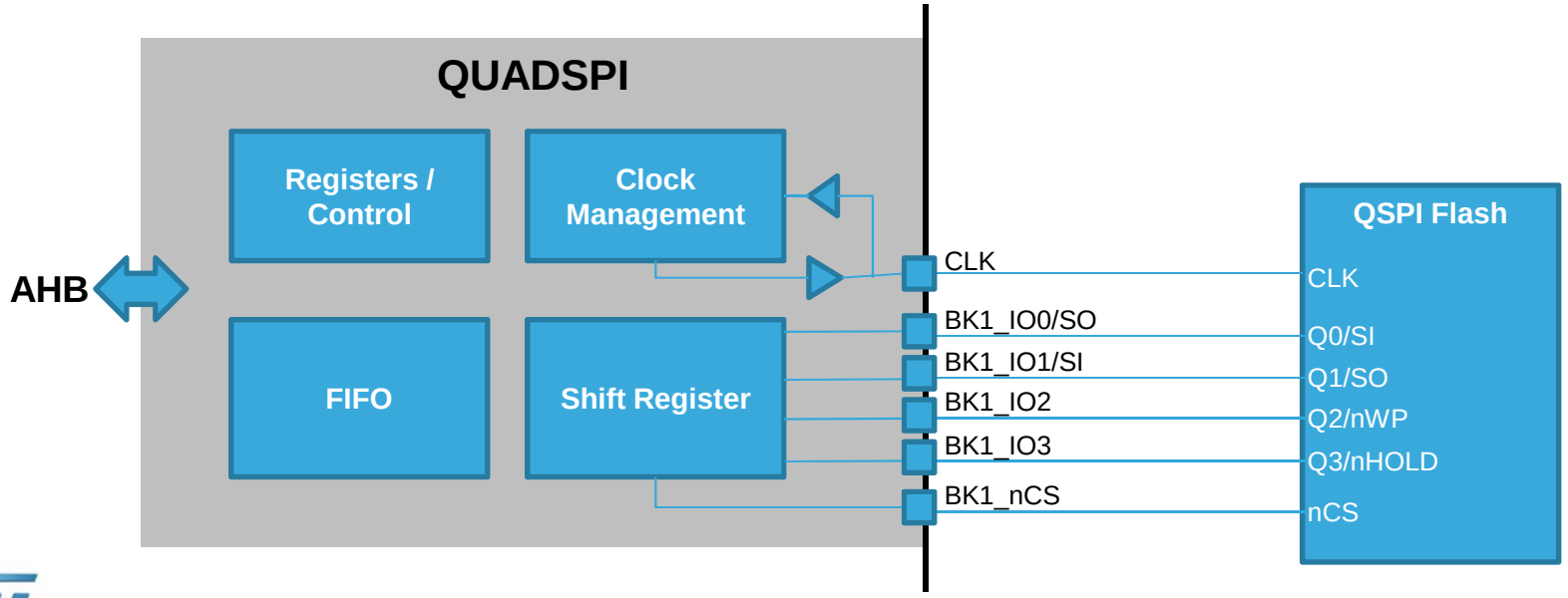


- Communication interface for single/dual/quad SPI flash memories

QuadSPI Overview

- Three operating modes
  - **Indirect** : All operations are performed through registers (classical SPI)
  - **Status polling**: Hardware polling of status registers (interrrupt generation)
  - **Memory mapped**: External flash seen as internal for read operations – 256MB limit

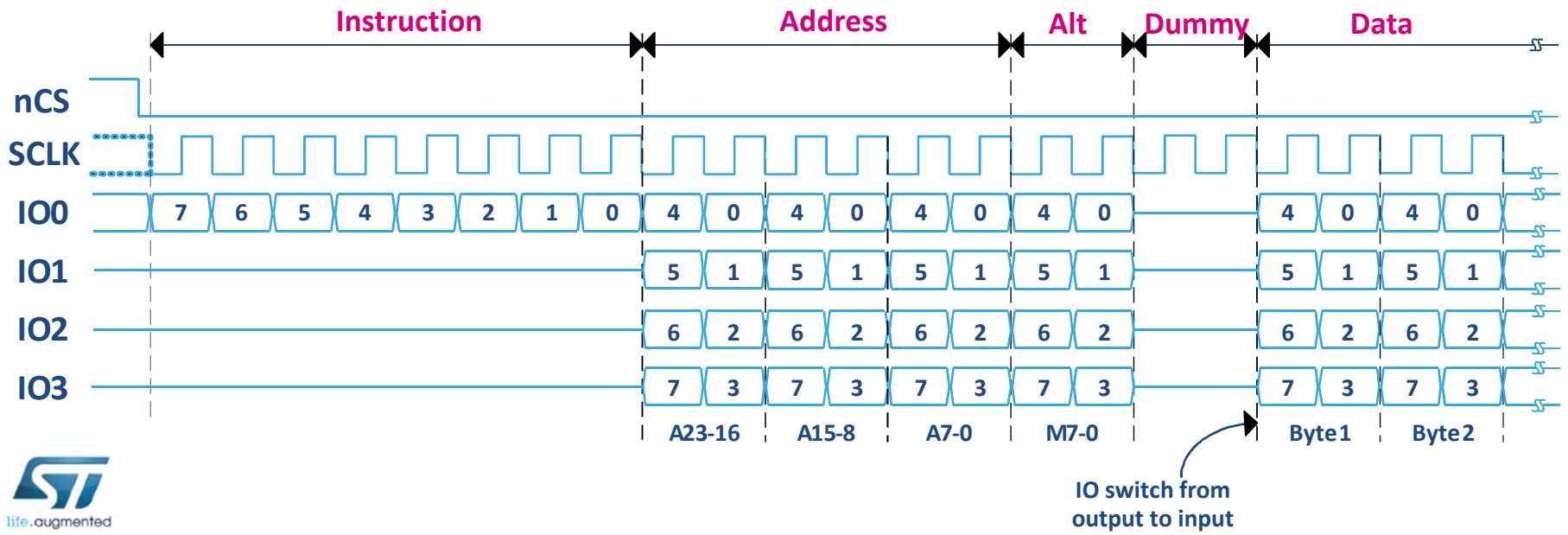
  - SDR and DDR support
  - DMA, Interrupts & Integrated FIFO for reception, transmission and error handling
  - 8, 16, and 32-bit data accesses



life.augmented

QuadSPI Frame format

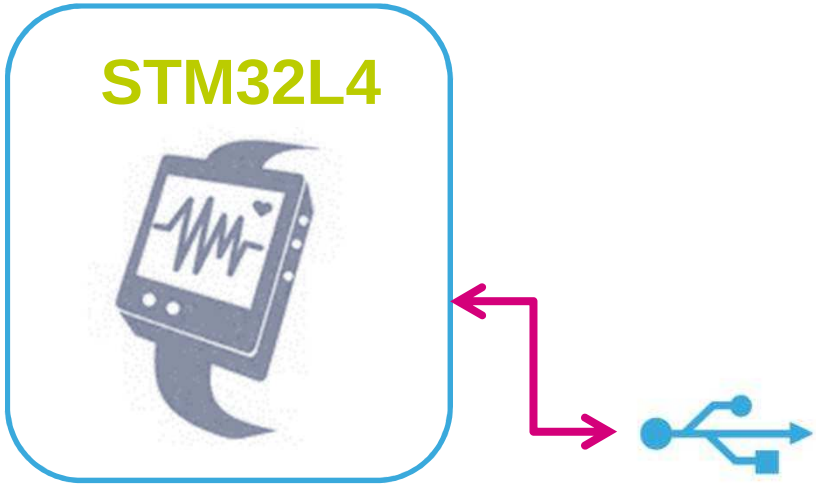
- Each of the 5 phases is fully configurable
  - Enabled or not
  - Length
  - Number of lanes
- Example of Read configuration
  - Instruction on 1 lane
  - Address, Alternate & Data on 4 lanes
  - 2 dummy cycles



life.augmented

Smart peripherals – USB OTG FS 2.0

- USB 2.0 / OTG 2.0 compliant
- VDDUSB: Dedicated 3.3V supply input
- 6 bidirectional endpoints
- MSI in PLL mode (auto-trimmed with LSE) to reach 48MHz & < 0.25% accuracy
  - HSE not needed
- Link power management (LPM) support
  - New power-save state, L1 (Sleep), with fast entry/exit compared to traditional L2 state (Suspend): **50uS** vs 10mS
- Battery charging detection (BCD) support
  - Detect and identify the port type (standard or charging)
- Attach Detection Protocol (ADP) support
  - Determines attachment status in the absence of bus power
- Suspend/resume support
  - Wakeup from STOP



Smart peripherals

Analog Peripherals

- OPAMP's:
  - Rail-to-rail inputs, 1mV offset after calibration
  - Outputs: 500µA (sink/source), 100µA (low-power mode)
  - 1MHz GBW (normal mode), 500kHz (low-power mode)
  - 0.7V/µs slew rate (normal), 0.3V/µs (low-power Mode)
- Comparators:
  - 2 Comparators
  - Window mode
  - Available in low-power modes
- DAC's:
  - 8/12-bit mode
  - Lots of conversion triggers
  - Programmable output buffer to drive more current
  - Supply: VDDA = 1.8 V to 3.6 V
  - **NEW!** Sample and hold low-power mode
- ADC's:
  - ADC1/ADC2 are tightly coupled, ADC3 is standalone
  - Consumption linear vs. conversion rate : 200 µA / MSPS
  - Dual-clock architecture
  - Up to 5.3Ms/s with 12-bit resolution in single mode
  - Single-ended or differential inputs
  - Internal channels: Temp sensor, VREF, VBAT/3, DAC
  - **NEW!** Hardware Oversampling

OPAMP
2x with built-in PGA

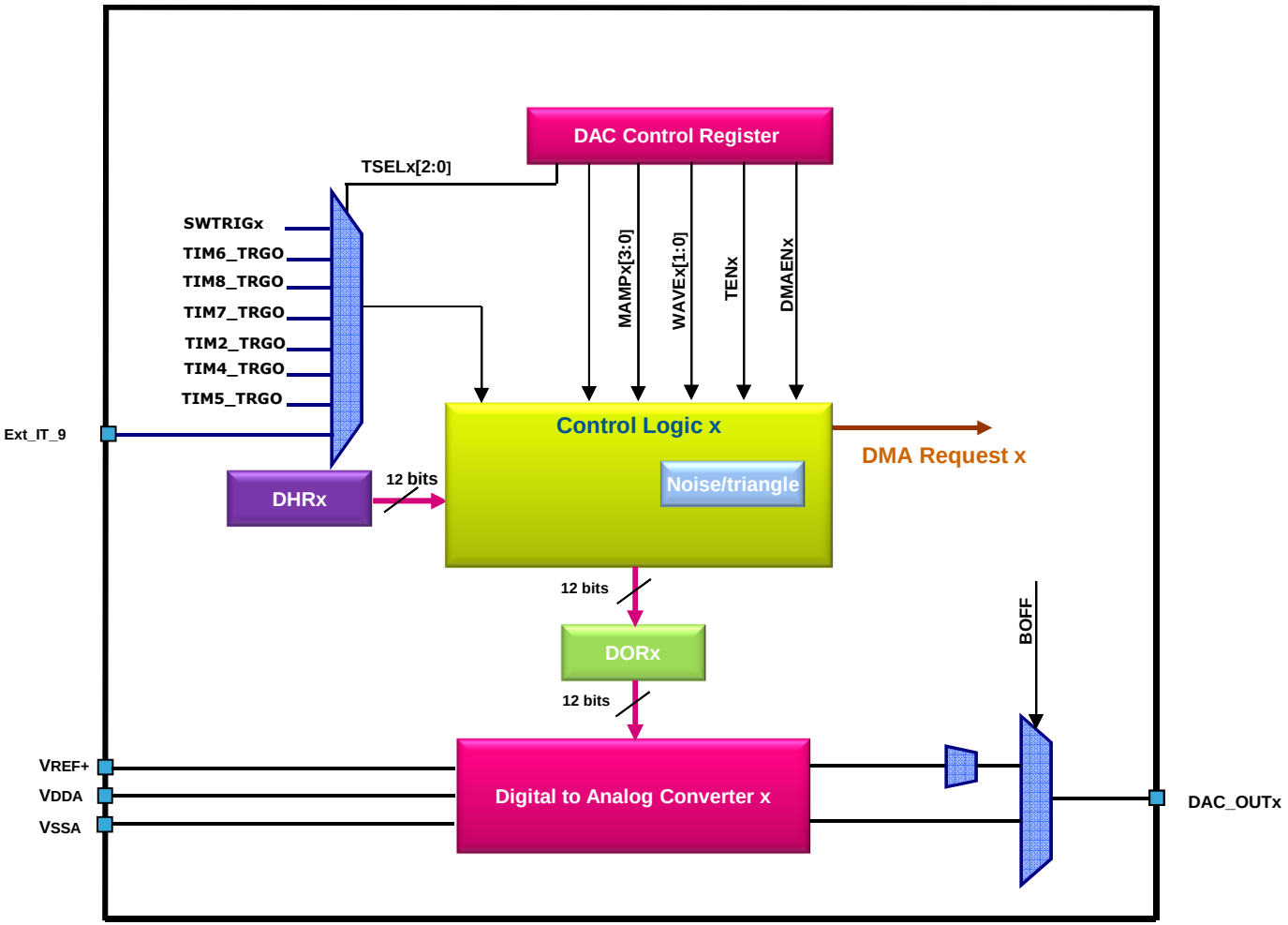
COMP
2x Low-power

DAC
2x Low-power sample and hold

ADC
3× 12-bit ADC 5 MSPS

life.augmented

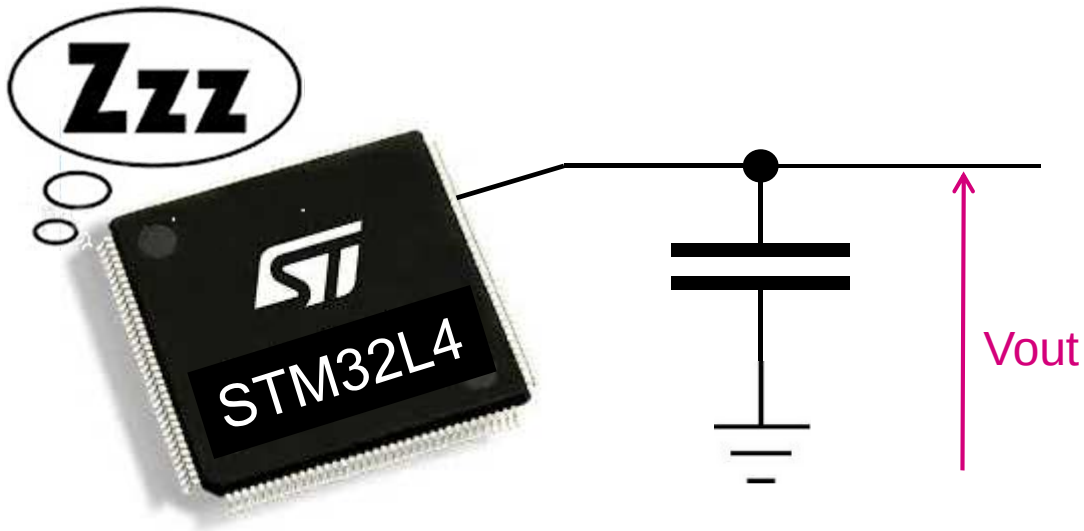
DAC1 / DAC2 block diagram



life.augmented

NEW! Sample & Hold feature

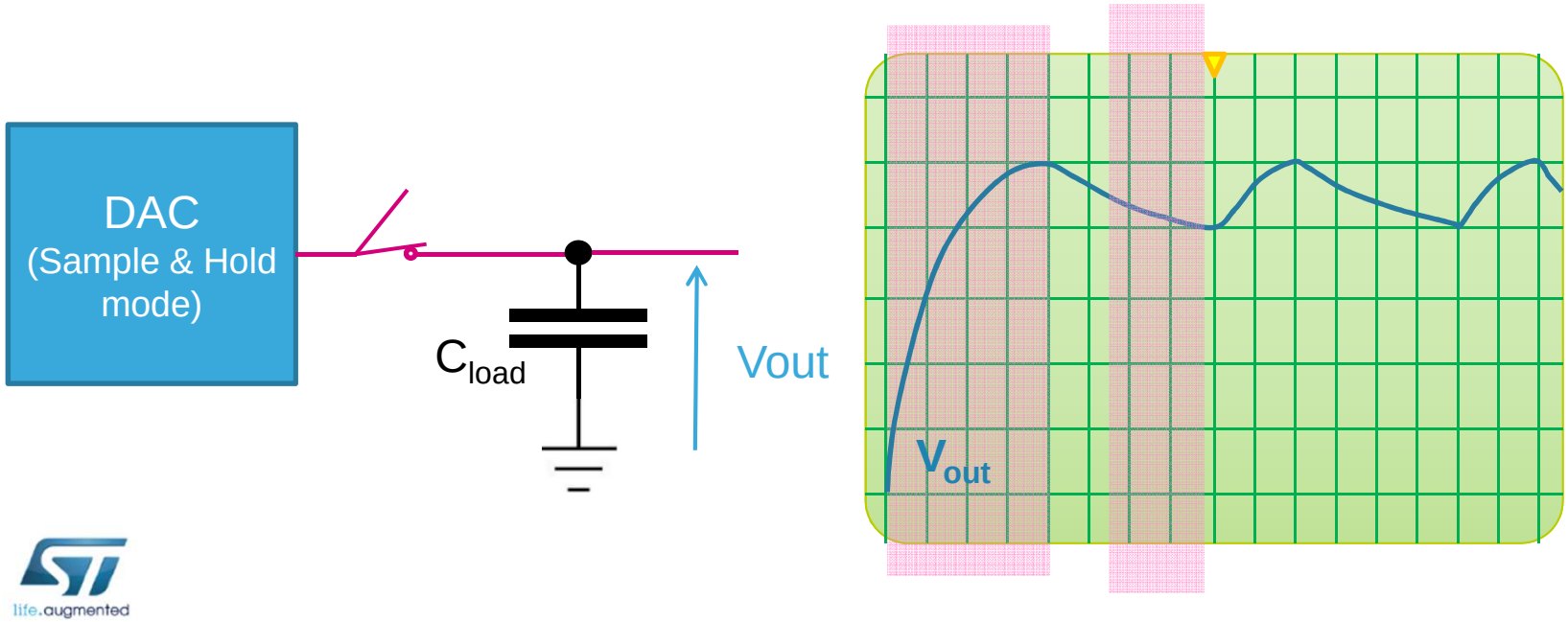
- Maintains DAC output voltage when the MCU is in a low power mode, such as STOP1 mode
- In ***Sample & Hold*** mode the DAC is able to hold its' output voltage while all related analog and digital blocks are shut off
- Up to 15x power savings in some configurations!



life.augmented

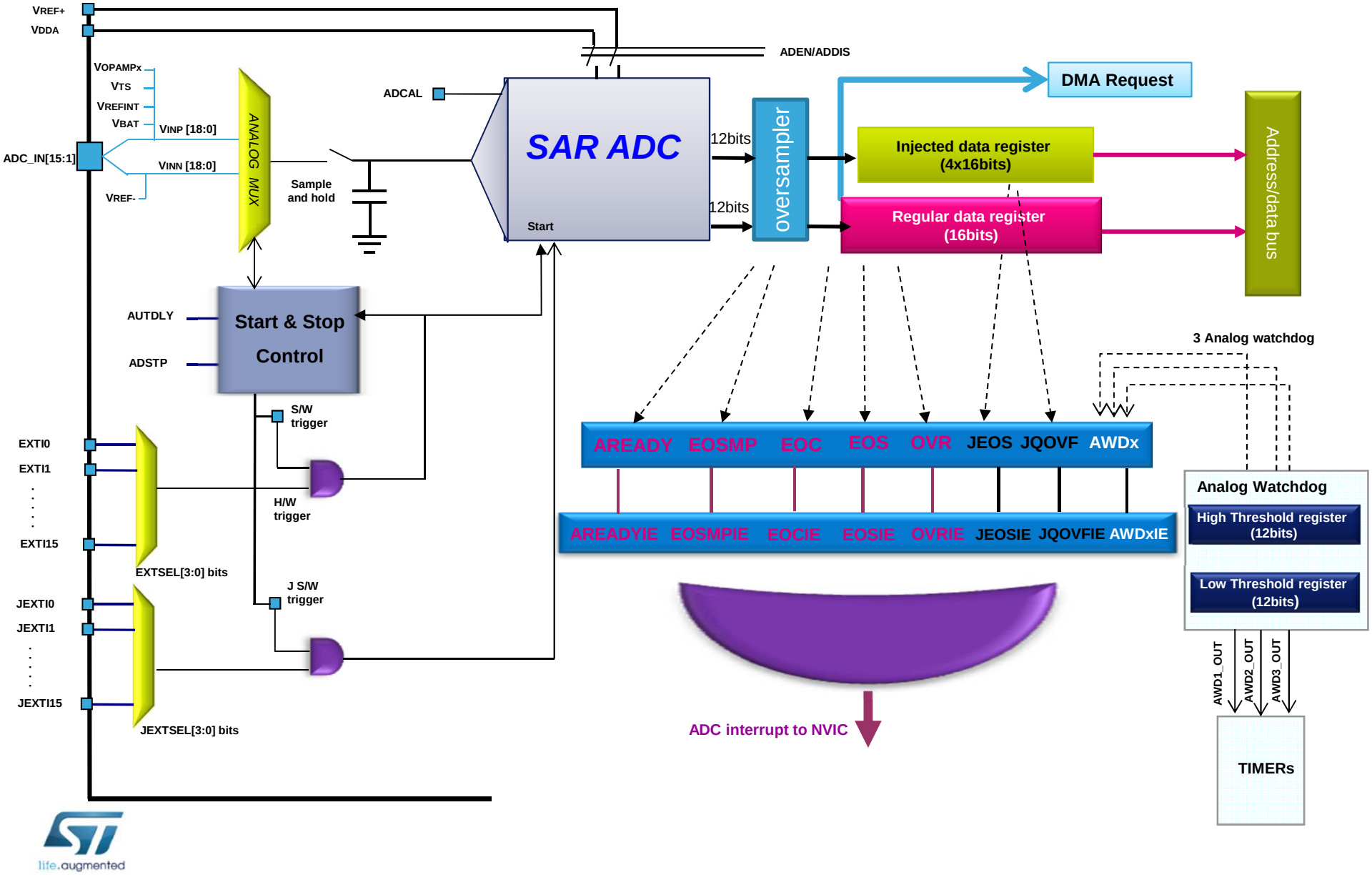
How it works?

- Sampling phase: During this phase, the “sample & hold” element is charged into the desired voltage.
- Holding phase: During which the DAC’s output is tri-stated (High-Z) to maintain the “Sample & Hold” element’s stored electrical charge.
- Refresh phase: Due to leakage coming from several sources, a refresh phase is essential to keep the output voltage at the desired value (+/- LSB)



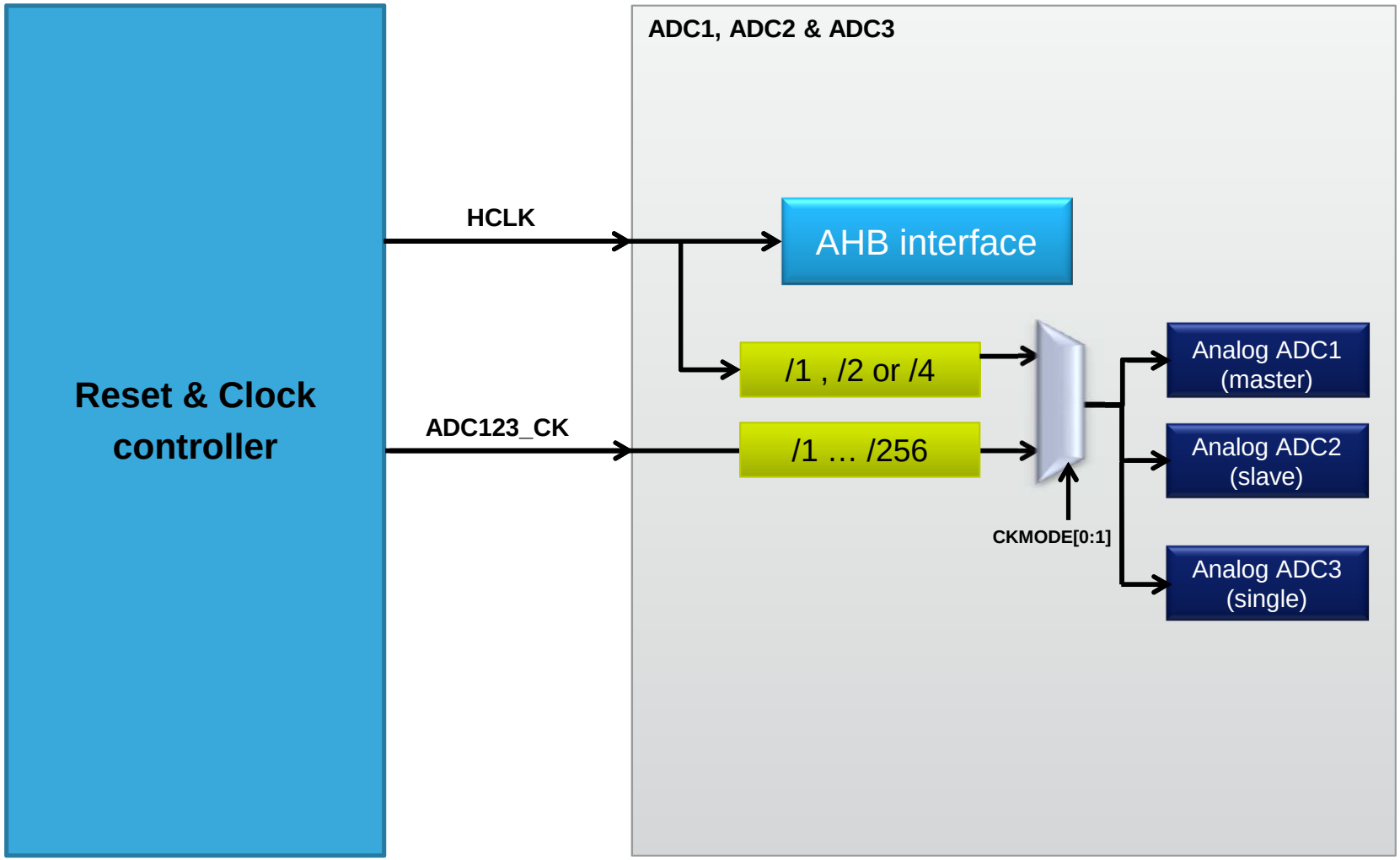
life.augmented

ADC Block Diagram



life.augmented

ADC Clocks

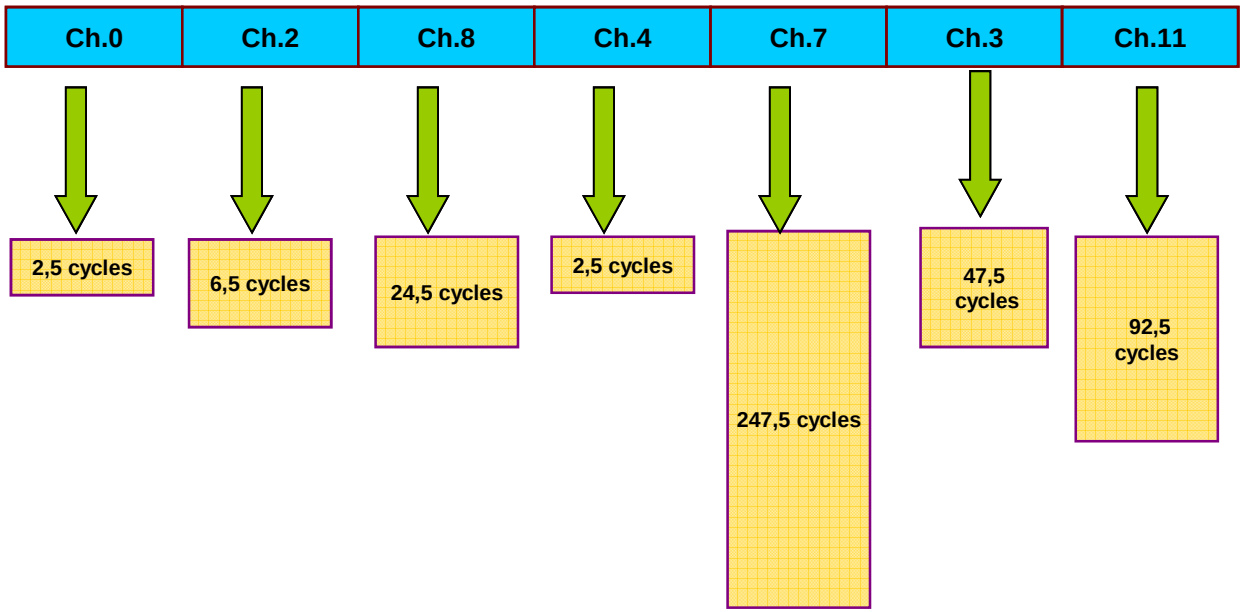

Reset & Clock controller
HCLK
ADC123_CK
ADC1, ADC2 & ADC3
AHB interface
/1 , /2 or /4
/1 … /256
CKMODE[0:1]
Analog ADC1 (master)
Analog ADC2 (slave)
Analog ADC3 (single)

ADC Sequencer

- Up to 16 regular and 4 injected conversions with programmable order and programmable sampling time,

Example: - Conversion of channels: 0, 2, 8, 4, 7, 3 and 11

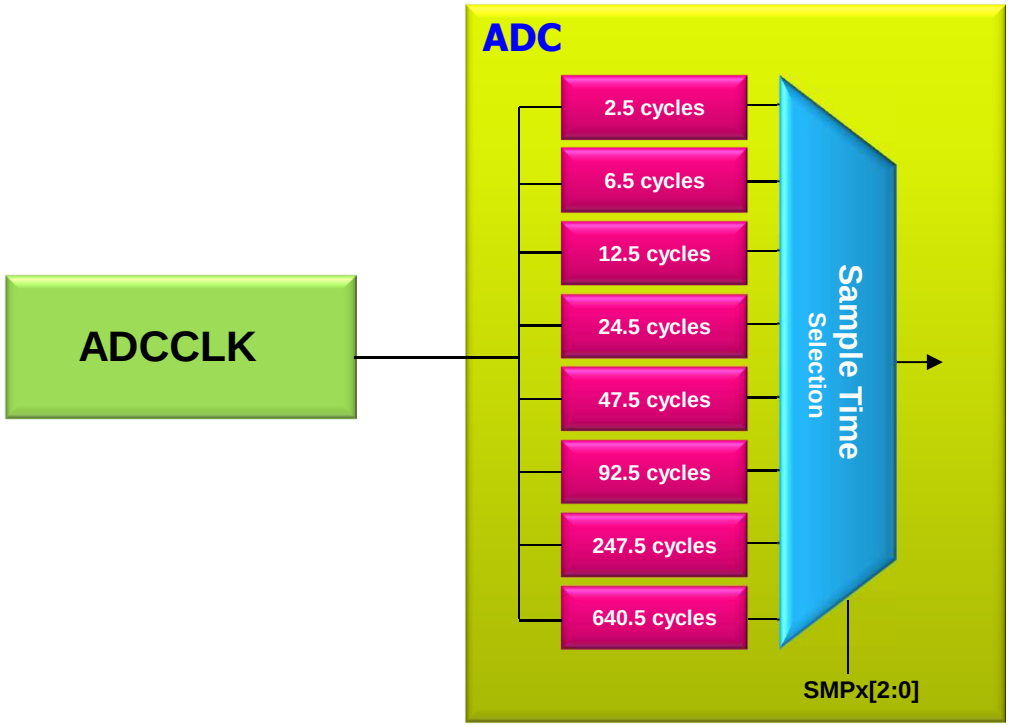
- Different sampling time.



life.augmented

ADC Sampling Time ($T_{Sampling}$)

- Three bits programmable sampling time channel by channel programmable:
  - 2.5 cycles
  - 6.5 cycles
  - 12.5 cycles
  - 24.5 cycles
  - 47.5 cycles
  - 92.5 cycles
  - 247.5 cycles
  - 640.5 cycles



Note: The sampling time value depends on the type of channel (fast or slow), the resolution and output impedance of the external signal source to be converted

life.augmented

Total Conversion Time

- Total conversion Time = TSampling + TConversion

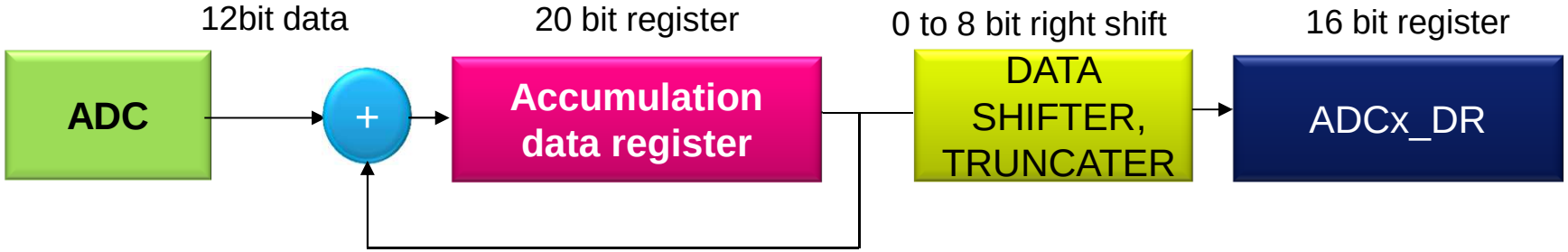
Resolution	$T_{Conversion}$
12 bits	12.5 Cycles
10 bits	10.5 Cycles
8 bits	8.5 Cycles
6 bits	6.5 Cycles

Resolution	Total conversion Time (When FADC = 80MHz)	
12 bits	12.5 + 2.5 = 15cycles	18.75 us → 5.33 Msps
10 bits	10,5 + 2.5 = 13 cycles	16.25 us → 6.15 Msps
8 bits	8.5 + 2.5 = 11 cycles	13.75 us → 7.27 Msps
6 bits	6.5 + 2.5 = 9 cycles	11.25 us → 8.89 Msps

life.augmented

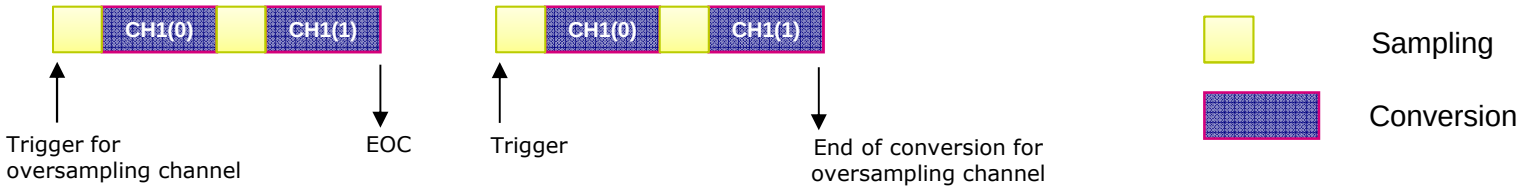
Oversampler

- Oversampler performs data pre-processing to offload the CPU. Handles up to 256 conversions and averages them
  - For : Averaging, Data rate reduction, SNR improvement, Basic filtering



- Programmable oversampling ratio
  - x2, x4, x8, x16, x32, x64, x128, x256
- Programmable data shifting / truncating
  - Right shift 0 to 8 bits

Example: Oversampling ratio x2



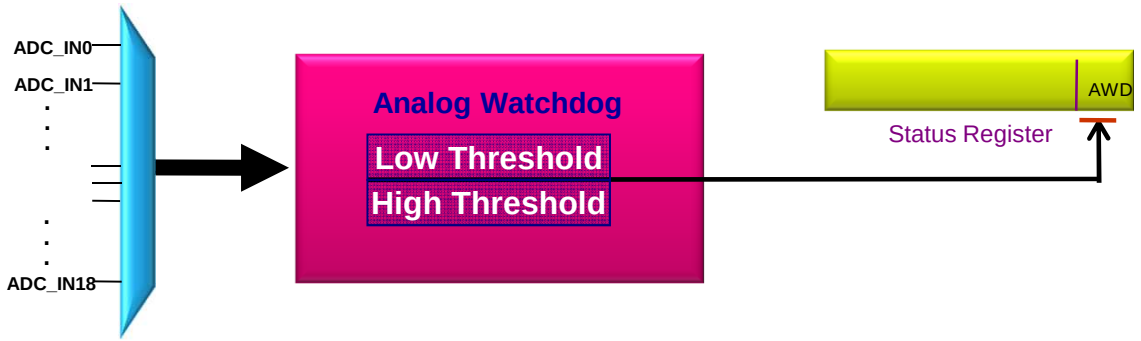
life.augmented

ADC Analog Watchdogs

- ADC Analog Watchdog 1
  - 12-bit programmable analog watchdog low and high thresholds
  - Enabled on one or all converted channels
  - Interrupt generation on low or high thresholds detection
- ADC Analog Watchdog 2&3

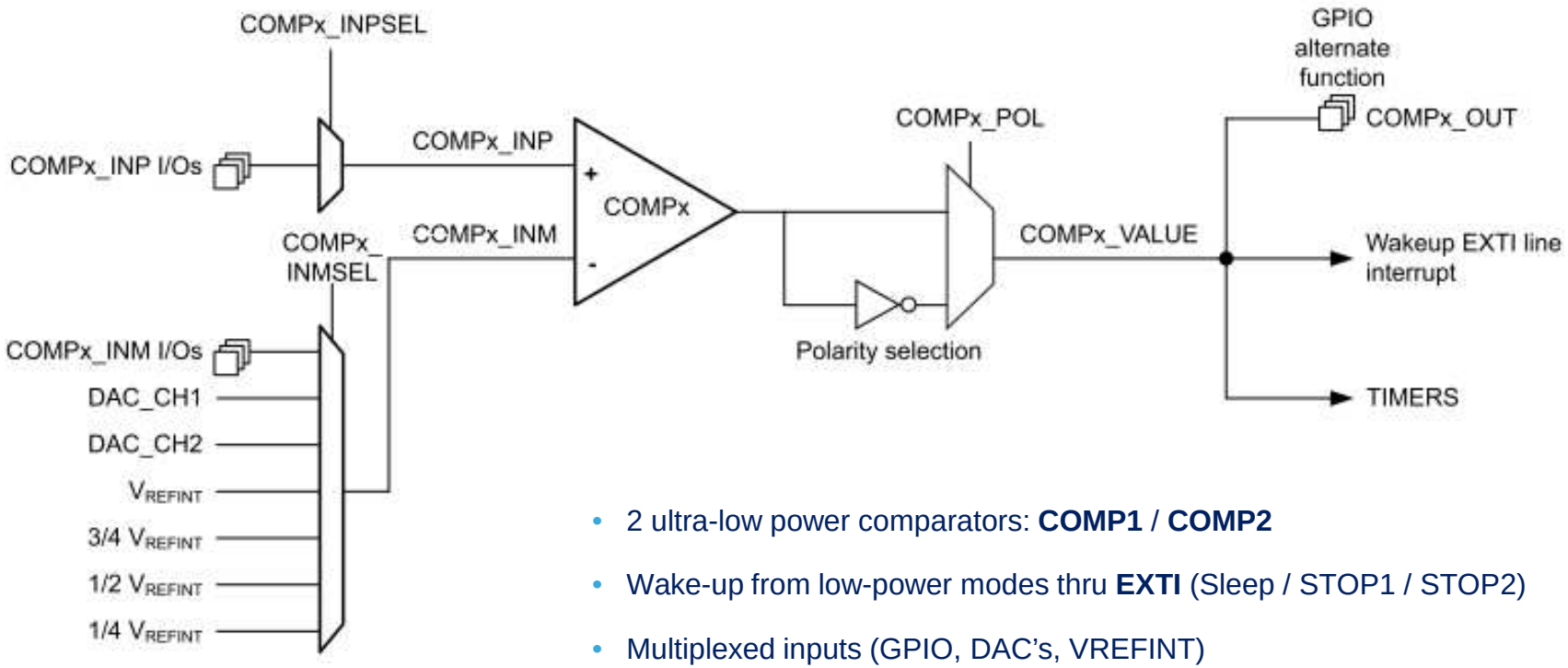
NEW!

  - Enabled on some selected channels by programming bits in AWDCHx[18:0],
  - Resolution Limited to 8 bits and only the 8 MSBs of the thresholds can be programmed



life.augmented

COMPARATORS



- 2 ultra-low power comparators: **COMP1** / **COMP2**
- Wake-up from low-power modes thru **EXTI** (Sleep / STOP1 / STOP2)
- Multiplexed inputs (GPIO, DAC's, VREFINT)
- Programmable **hysteresis and speed-vs-consumption**
- Redirection of output to IO or timer inputs (e.g. TIM Break event)
- Outputs with **blanking** source
- Comparators can be combined in the **window** comparator

life.augmented

COMP low power features

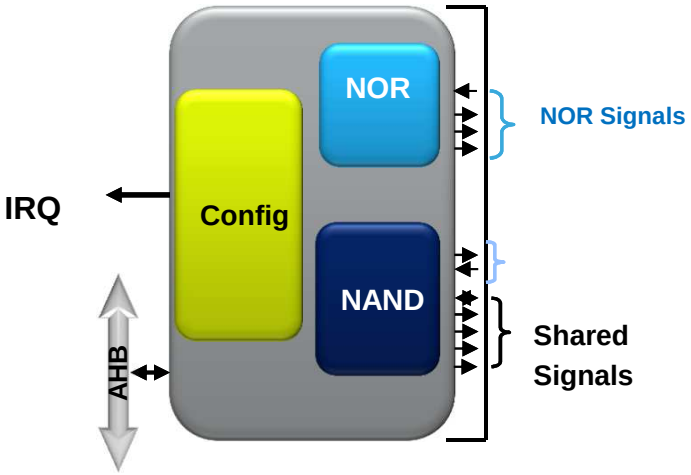
Power consumption -vs- propagation delay can be adjusted:

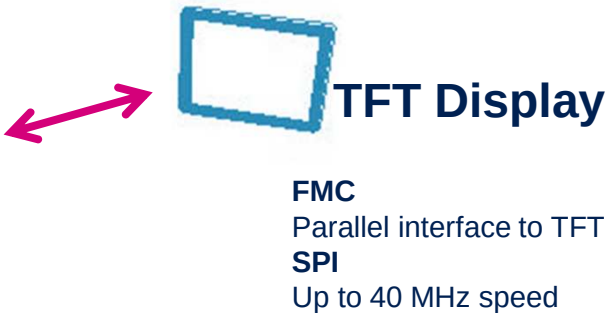
PWRMODE	Max propagation delay	Consumption (Typ)
00	80 ns	70 µA
01 or 10	1 µs	5 µA
11	12 µs	350 nA

LP mode	Feature / limitation
Run, LPRun	no limitation, state polling or interrupt thru EXTI
Sleep, LPSleep	wakeup capability thru EXTI
Stop 1, Stop 2	wakeup capability thru EXTI
Standby	not available
Shutdown	not available

life.augmented

Smart peripherals

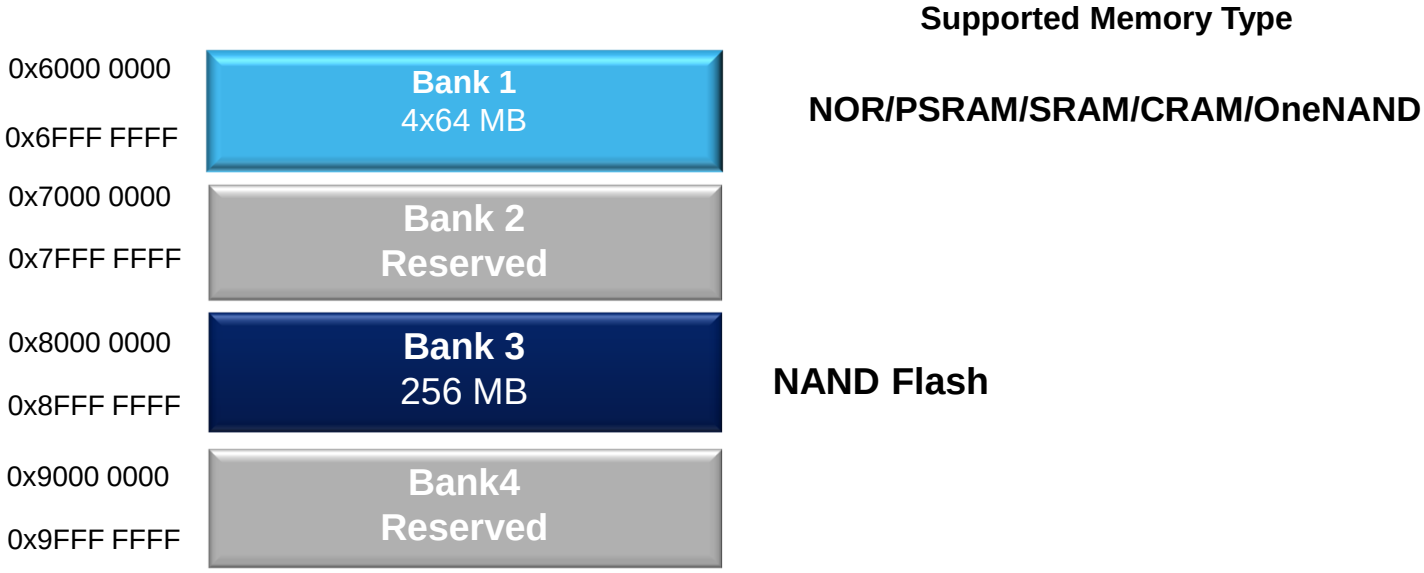




- NOR Flash/PSRAM and NAND controllers
- Differences from FSMC:
  - Continuous FMC clock generation for synchronous and asynchronous modes.
  - Performance enhancement
  - Removal of PCCard controller on new products.

FMC Bank memory mapping

- For the FMC, the external memory is divided into 4 fixed size banks of 4x64 MB each:
  - Bank 1 can be used to address NOR Flash, OneNAND or PSRAM memory devices.
  - Banks 3 is used to address NAND Flash devices.
  - Bank 2 & 3 reserved



LPTIM Features Summary

- Asynchronous running capability

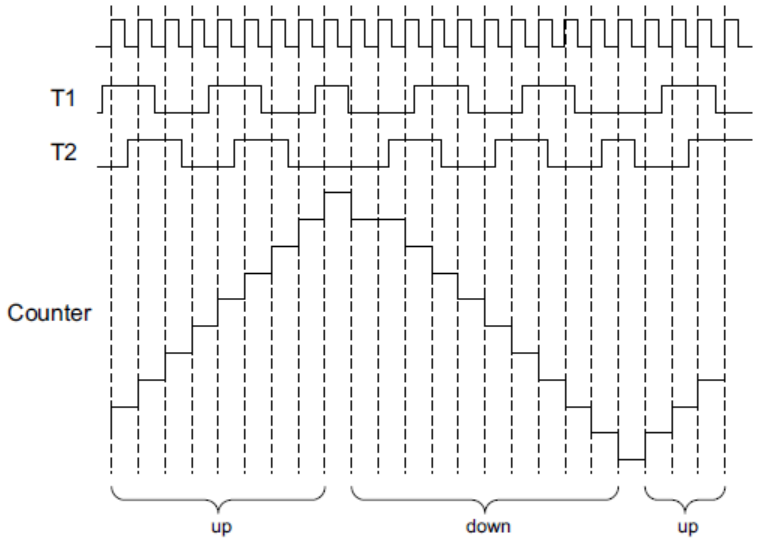
- Ultra low power-consumption

- Timeout function for wakeup from Stop 1 mode (LPTIM1 & LPTIM2) and Stop 2 mode (LPTIM1 only)

life.augmented

LPTIM Features

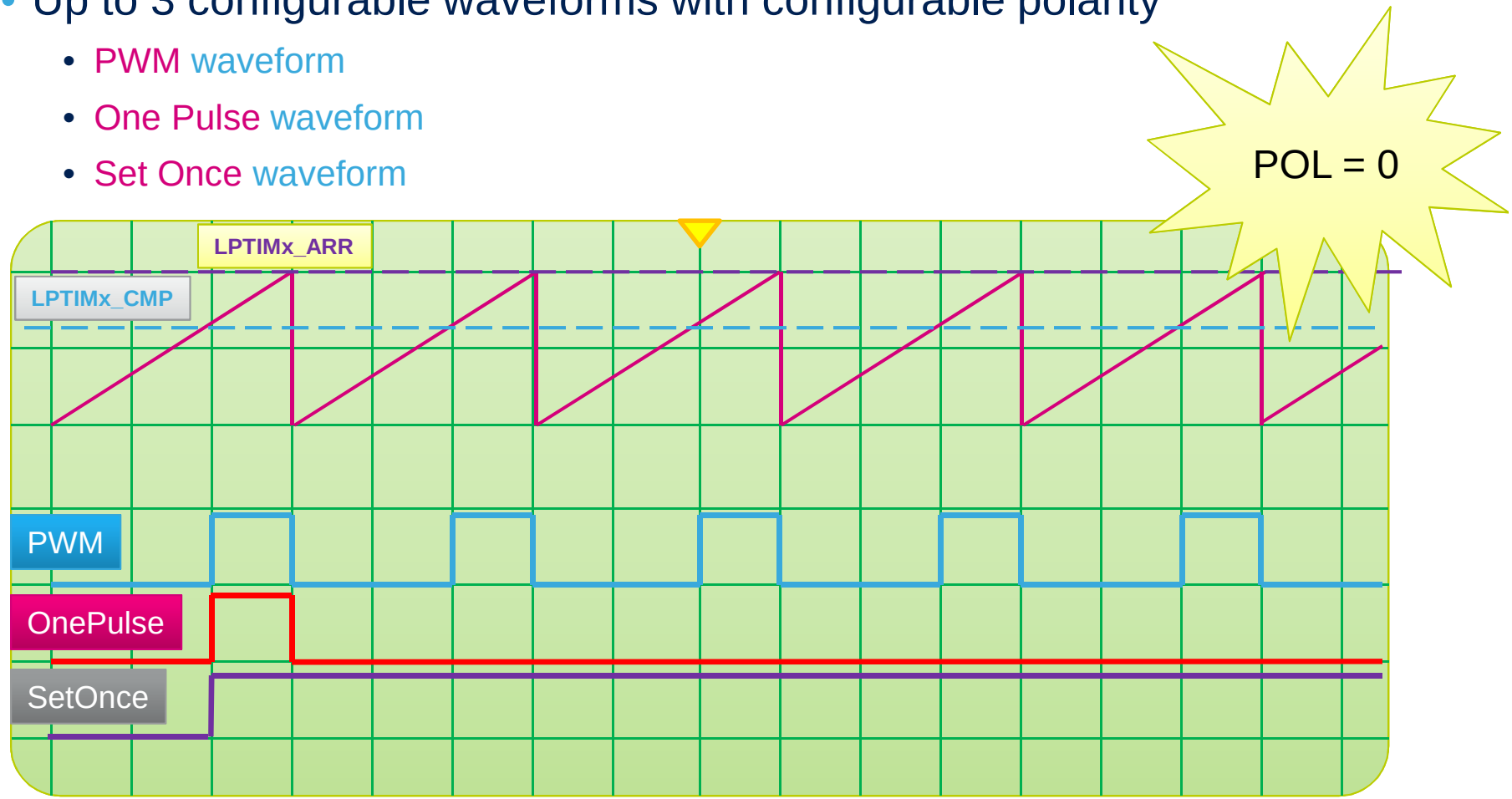
- Up to 5 clock sources to achieve lowest power consumption
  - APB clock, LSE, LSI, HSI, External clock
- Internal / External hardware triggers, with digital glitch filter:
  - Rising / Falling or Both edges
  - GPIO, RTC events, COMP1/2
- Up to 2 operation modes
  - **Continuous mode**: free running mode; many counter overruns are possible
  - **One Shot mode**: Counter stops counting when the overrun value is reached
- Encoder mode (LPTIM1 only)
- 6 interrupt sources



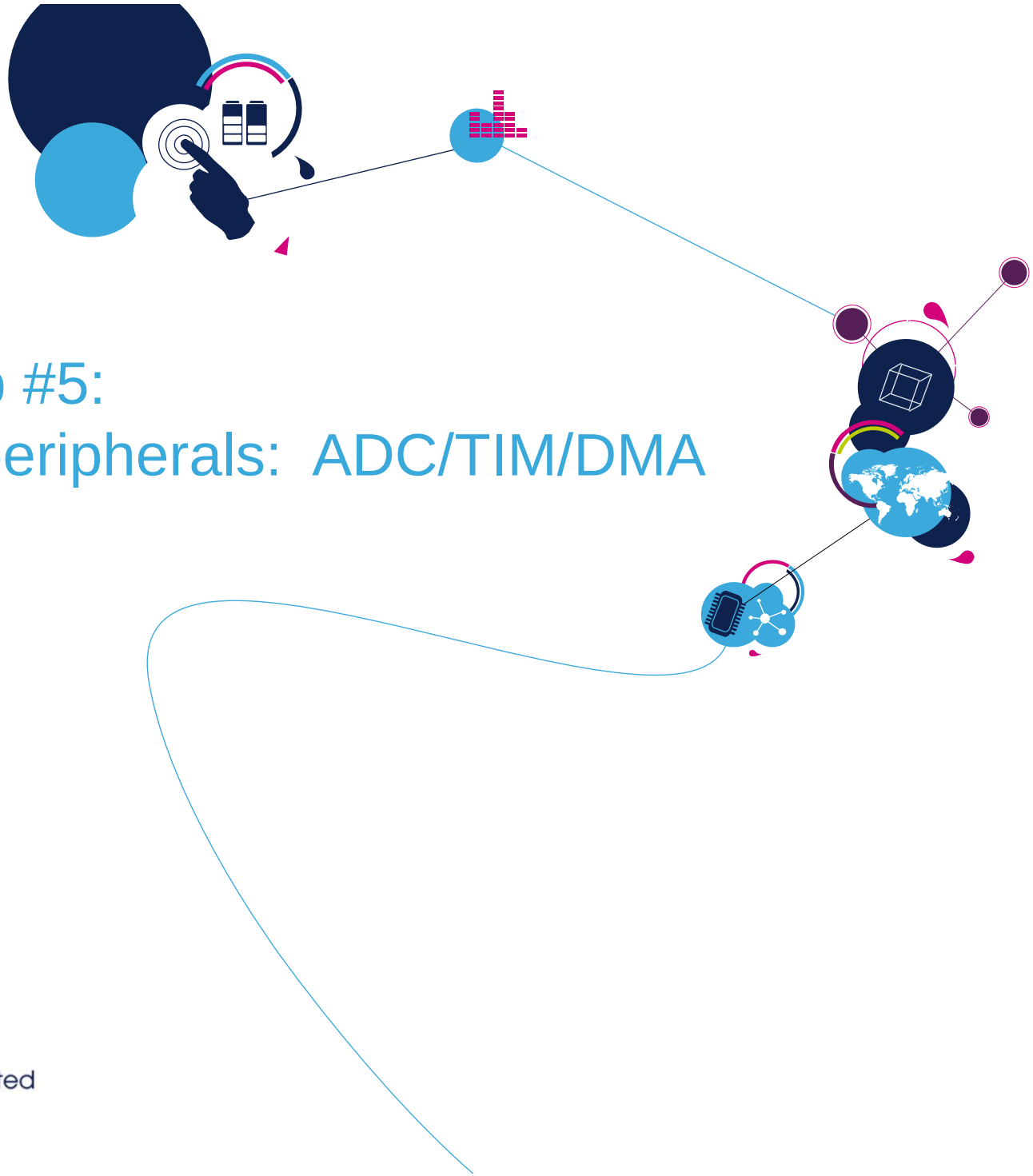
life.augmented

LPTIM Features

- Up to 3 configurable waveforms with configurable polarity
  - PWM waveform
  - One Pulse waveform
  - Set Once waveform



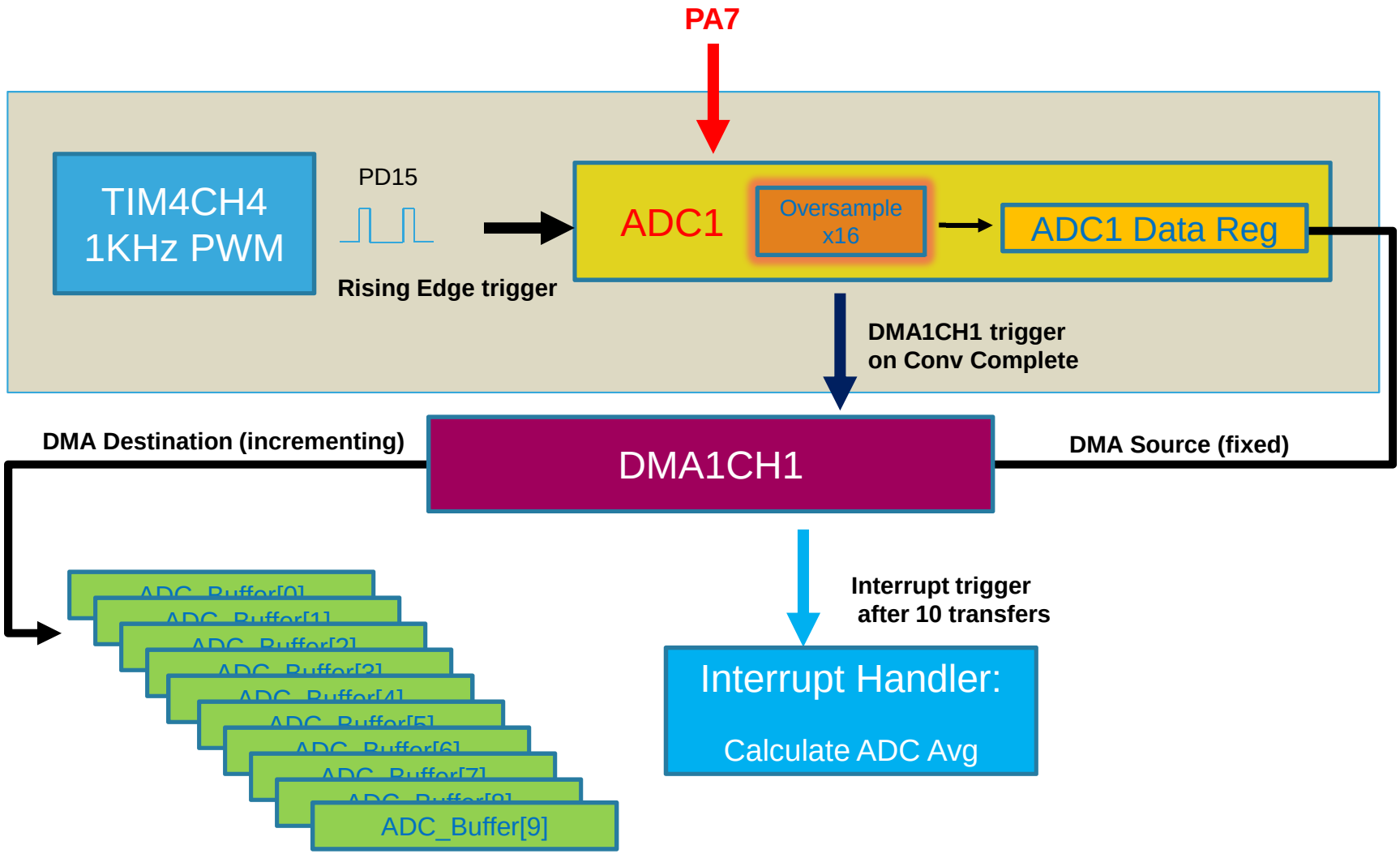
life.augmented



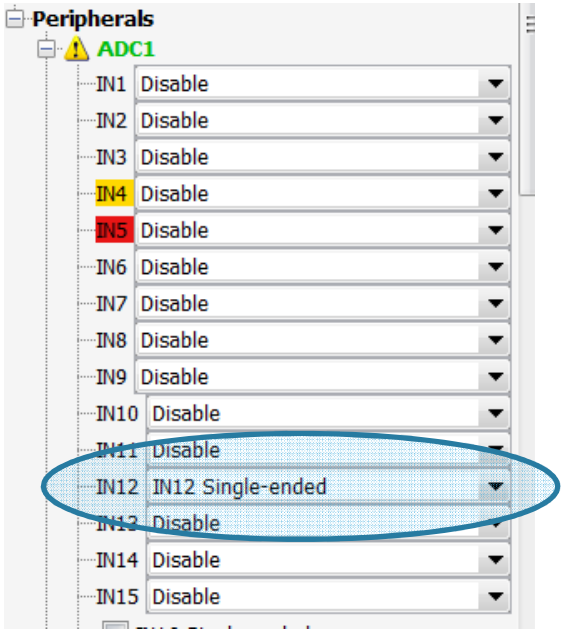
Hands-On Lab #5:
Autonomous peripherals: ADC/TIM/DMA

life.augmented

Hands-On Lab #5: ADC-TIM-DMA functionality



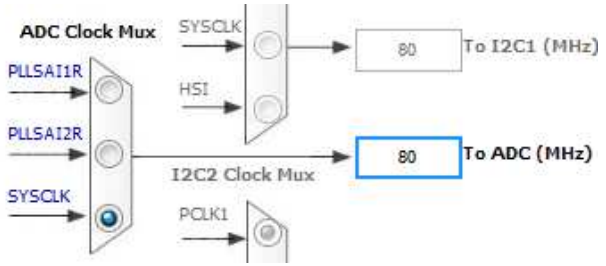
ADC1 GPIO / Clock setup:



Set up ADC GPIO / Clocks:

PA7 – ADC1_IN12 – Analog Input, Single-Ended

ADC Clock Source = SYSCLK (80MHz)



life.augmented

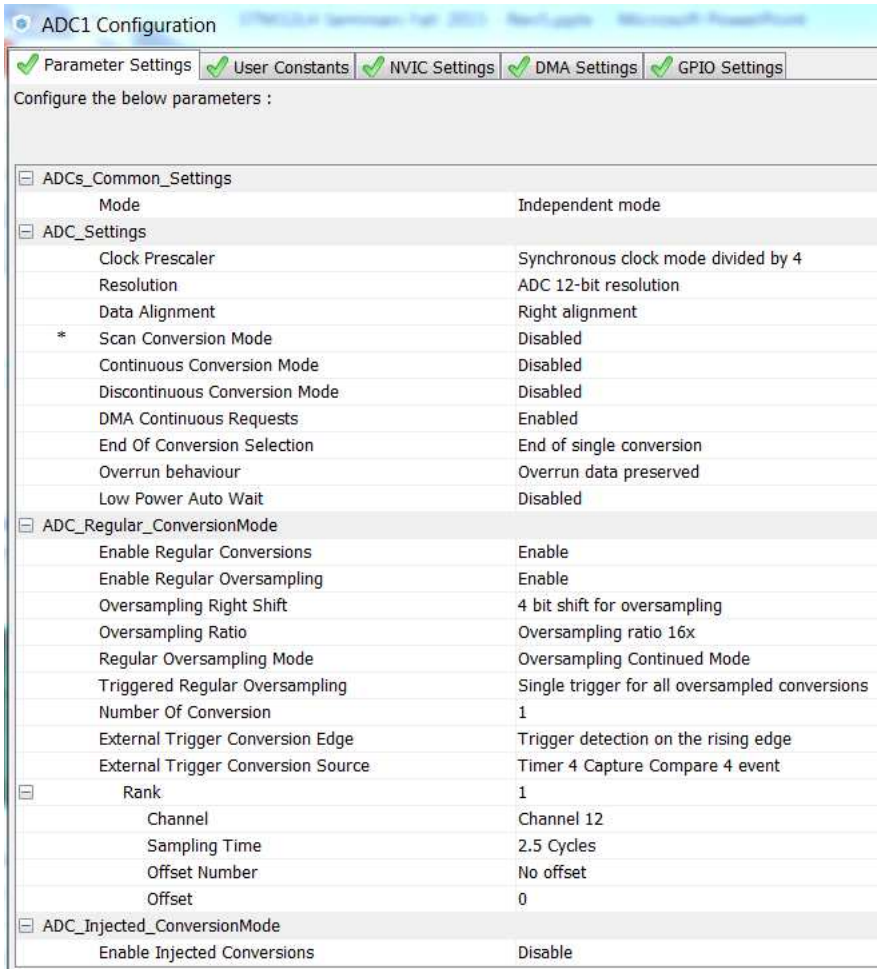
ADC1 Settings:

Changes from default settings:

Synchronous clock mode divided by 4
DMA Requests Enabled, End of Single conversion

Enable Regular Oversampling
4-bit shift
Oversampling ratio 16x

External Trigger Conversion Edge: Rising Edge
External Trigger Conversion Source: TIM4 CapCom 4



life.augmented

ADC1 DMA and DMA interrupt:

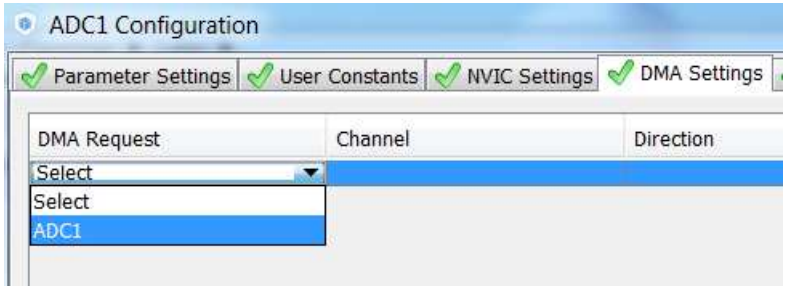
Select DMA Settings tab in ADC1 Configuration:

Click “Add” button

Select ADC1

Use Word Data Width for Peripheral and Memory

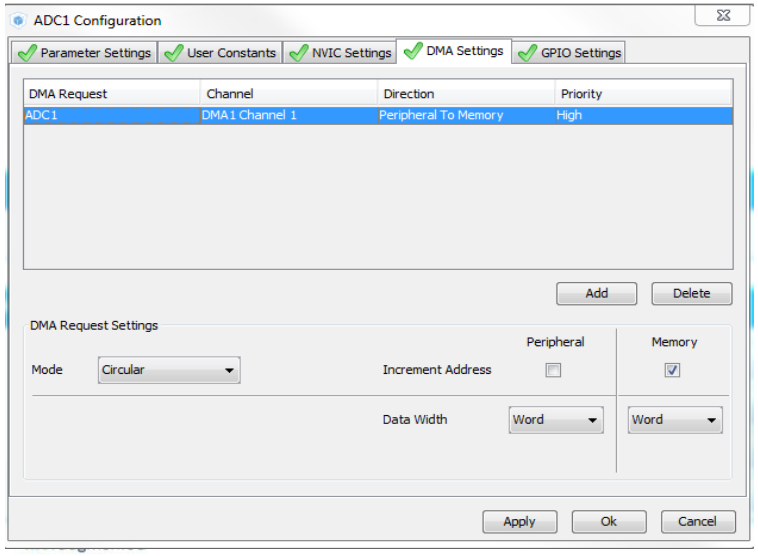
Increment Memory Address

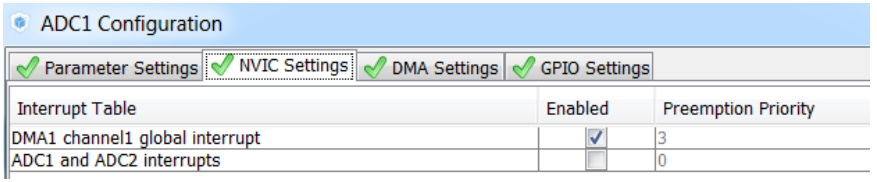


Select NVIC Settings tab:

Make sure DMA1 channel1 global interrupt is Enabled

Click “OK”





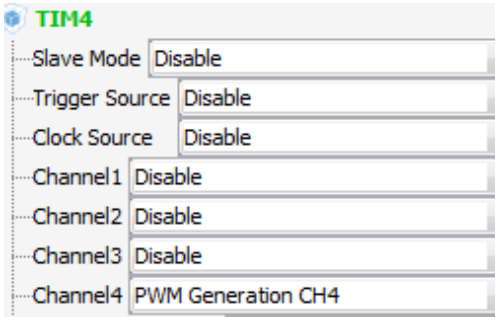
TIM4 CH4 setup:

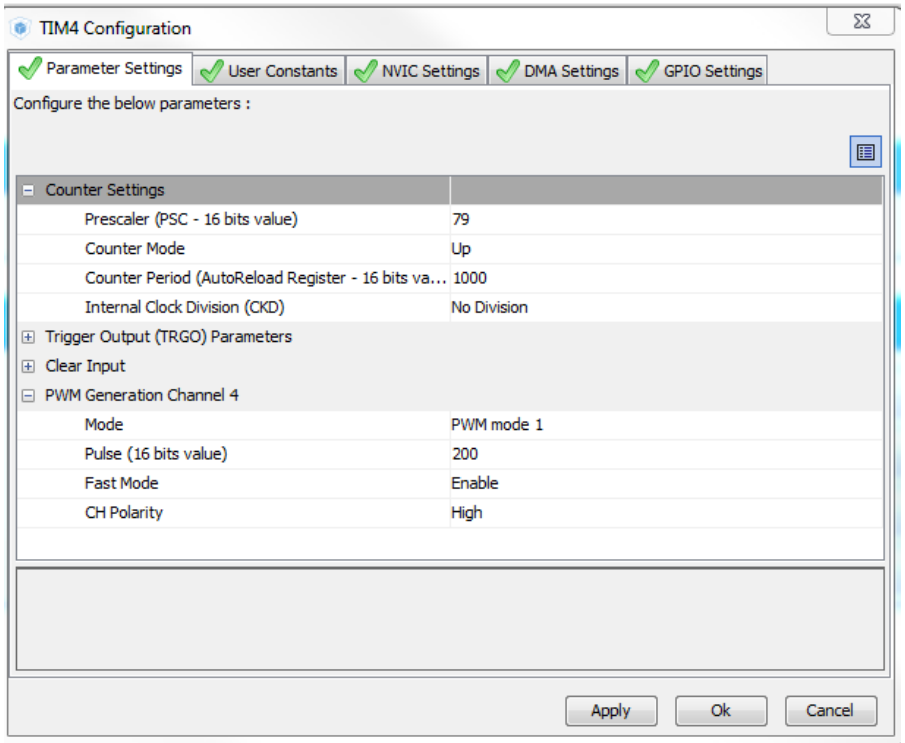
Set up GPIO:
PD15 – TIM4CH4 Alt Fn output – PWM Generation CH4

TIM4 settings for 1KHz, 20% duty-cycle PWM output:
Prescaler = 79 (79+1=80, 80MHz / 80 = 1MHz timer tick)
Counter Mode = UP
Counter Period = 1000 (1MHz / 1000 = 1KHz)
No Internal Clock Division

PWM Mode 1
Pulse = 200
Fast Mode: Enabled
CH Polarity: High

User Code HAL function calls required:
C:\STM32L4Seminar\Labs**lab5_adc_tim_dma.c**





life.augmented

Hands-On Lab #5: ADC-TIM-DMA functionality

Internal L4 ADC / TIM / DMA peripherals

Set up additional GPIO / Clocks:
PA7 – ADC1_IN12 – Analog Input, Single-Ended
ADC Clock = SYSCLK/4 (20MHz)
PD15 – TIM4CH4 Alt Fn output – PWM Generation CH4

ADC1 settings:
Clock Prescaler = Synch /4
12-bit, Right aligned, No Scan, Scan/Continuous modes disabled
x16 oversampling, 4-bit shift, continued mode
Trigger on TIM4CC4, Rising edge
Enable DMA, 1 conversion
Channel 12: Rank1, 2.5 cycle sampling, no offset

DMA settings:
DMA1CH1, Circular mode, Increment memory address, Word data width
DMA1CH1 interrupt enabled, preempt priority = 3

TIM4 settings:
Counting UP, Prescaler=79, Counter Period = 1000, No CKD
PWM Mode 1, Pulse = 200, Fast Mode enabled, Polarity High

User Code HAL function calls required:
C:\STM32L4Seminar\Labs**lab5_adc_tim_dma.c**

Lab#5: Autonomous Peripherals

- Needed code bits from C:\STM32L4Seminar\Labs**lab5_adc_tim_dma.c** into main.c:
  - **Add code bits between:** **/* USER CODE BEGIN PV */** **/* USER CODE END PV */**
  - **Add code bits between:** **/* USER CODE BEGIN 2 */** **/* USER CODE END 2 */**
  - **Add code bits between:** **/* USER CODE BEGIN WHILE */** **/* USER CODE END WHILE */**
  - **Add code bits between:** **/* USER CODE BEGIN 4 */** **/* USER CODE END 4 */**

  - Open a terminal emulator, using USART2 settings, Virtual COM port xx
  - Rebuild, Program/Debug, Run!

  - On the rising edge of each TIM4 PWM cycle, the ADC is triggered.
  - The ADC input is oversampled 16 times
  - The summed result is shifted right 4 places, performing a hardware averaging
  - A DMA transfer moves this result into a data array and the array pointer is incremented
  - 10 data elements are transferred and then the cycle repeats

- Add the array **ADC_Result** to the Live Watch debug window to see updates!

SUMMARY 4 Keys of STM32 L4 series

1. ULP leader and performance booster
2. Innovation
3. Integration and safety
4. Great Investment

life.augmented

STM32 apps & social media

- Find more about STM32 products and solutions:

ST MCU Finder mobile application

www.st.com/stmcufinder

Social media

ST Forums on microcontrollers

facebook.com/stm32

youtube.com/STonlineMedia

twitter.com/@ST_World

Mbed.org

ARM mbed

ARM Connected Community

STM32 @ ARM connected community

life.augmented

Thank you

/STM32 @ST_World st.com/e2e

www.st.com/stm32l4

life.augmented